ENHANCED EVOLUTIONARY ALGORITHM WITH CUCKOO SEARCH FOR NURSE SCHEDULING AND RESCHEDULING PROBLEM

LIM HUAI TEIN

DOCTOR OF PHILOSOPHY
UNIVERSITI UTARA MALAYSIA
2015

Awang Had Salleh
Graduate School
of Arts And Sciences

Universiti Utara Malaysia

PERAKUAN KERJA TESIS / DISERTASI
(Certification of thesis / dissertation)

Kami, yang bertandatangan, memperakukan bahawa
(We, the undersigned, certify that)

LIM HUAI TEIN

calon untuk Ijazah PhD
(candidate for the degree of)

telah mengemukakan tesis / disertasi yang bertajuk:
(has presented his/her thesis / dissertation of the following title):

"ENHANCED EVOLUTIONARY ALGORITHM WITH CUCKOO SEARCH FOR NURSE SCHEDULING AND RESCHEDULING PROBLEM"

seperti yang tercatat di muka surat tajuk dan kulit tesis / disertasi.
(as it appears on the title page and front cover of the thesis / dissertation).

Bahawa tesis/disertasi tersebut boleh diterima dari segi bentuk serta kandungan dan meliputi bidang ilmu dengan memuaskan, sebagaimana yang ditunjukkan oleh calon dalam ujian lisan yang diadakan pada : *21 September 2015.*
That the said thesis/dissertation is acceptable in form and content and displays a satisfactory knowledge of the field of study as demonstrated by the candidate through an oral examination held on: September 21, 2015.

Pengerusi Viva: *(Chairman for VIVA)* Assoc. Prof. Dr. Sharipah Soaad Syed Yahaya Tandatangan *(Signature)*

Pemeriksa Luar: *(External Examiner)* Prof. Dr. Rosalina Abdul Salam Tandatangan *(Signature)*

Pemeriksa Dalam: *(Internal Examiner)* Dr. Syariza Abd Rahman Tandatangan *(Signature)*

Nama Penyelia/Penyelia-penyelia: *(Name of Supervisor/Supervisors)* Assoc. Prof. Dr. Razamin Ramli Tandatangan *(Signature)*

Tarikh:
(Date) *September 21, 2015*

Permission to Use

In presenting this thesis in fulfilment of the requirements for a postgraduate degree from Universiti Utara Malaysia, I agree that the Universiti Library may make it freely available for inspection. I further agree that permission for the copying of this thesis in any manner, in whole or in part, for scholarly purpose may be granted by my supervisor(s) or, in their absence, by the Dean of Awang Had Salleh Graduate School of Arts and Sciences. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain shall not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Universiti Utara Malaysia for any scholarly use which may be made of any material from my thesis.

Requests for permission to copy or to make other use of materials in this thesis, in whole or in part, should be addressed to:

Dean of Awang Had Salleh Graduate School of Arts and Sciences
UUM College of Arts and Sciences
Universiti Utara Malaysia
06010 UUM Sintok

Abstrak

Kekurangan jururawat, ketidakhadiran yang tidak menentu dan stres mengundang persekitaran kerja yang tidak sihat di hospital. Isu ini memberi impak ke atas kehidupan sosial jururawat dan kesilapan dalam pemberian ubat yang boleh mengancam keselamatan pesakit. Impak ini telah menyumbang kepada peletakan jawatan para jururawat dan perkhidmatan kejururawatan yang berkualiti rendah. Bagi menangani isu ini, pengagihan jururawat yang sedia ada secara maksima melalui penjadualan kerja jururawat adalah alternatif terbaik. Namun, terdapat masalah pembinaan jadual kerja yang tidak memuaskan dan tidak stabil. Justeru, kajian ini mengintegrasikan komponen penjadualan dan penjadualan semula jururawat yang lazimnya diselesaikan secara berasingan dalam kajian lepas. Namun, bila mana perubahan spontan terhadap jadual dan bilangan kekangan yang perlu dipatuhi, didapati terdapat kekurangan unsur fleksibiliti dalam sebahagian besar kaedah penjadualan dan penjadualan semula. Dengan menerapkan unsur fleksibiliti, ia memberi satu platform yang berpotensi untuk menambah baik Algoritma Evolusi (EA), yang juga dikenal pasti sebagai kaedah penyelesaian. Demi meminimumkan pelanggaran kekangan dan membuat perubahan minimum yang berkesan ke atas jadual yang dipostulatkan semasa gangguan berlaku, model EA dengan kaedah Carian Kedasih (CS) dicadangkan. Suatu konsep enzim sekatan telah disesuaikan dalam CS. Sejumlah 11 model varian EA dibina dengan tiga operator pemilihan induk baharu, dua operator penyilangan baharu dan satu operator pembaikan berasaskan penyilangan, yang kesemuanya adalah sumbangan teoretikal. Keputusan kajian mendapati model EA dengan Penyilangan Pertandingan Kadar Penemuan dan Titik Enzim Sekatan Carian Kedasih (D_rT_CSREP) adalah paling berkesan dalam menghasilkan 100% jadual mampu berfungsi dengan nilai penalti paling minimum. Tambahan pula, semua gangguan jadual yang diuji telah berjaya diselesaikan melalui operator pra-pembaikan dan operator Pembaikan Titik Enzim Sekatan Carian Kedasih ($CSREP_r$). Hasilnya, model EA yang dibina mampu memenuhi kehendak para jururawat, menawar jadual atas panggilan yang adil, penjadualan semula yang lebih berkualiti dalam pertukaran syif, dan kefahaman tentang kebergantungan dua hala antara penjadualan dan penjadualan semula dengan mengambil kira keseriusan gangguan dalam penjadualan.

Kata Kunci: Algoritma evolusi hibrid, Carian Kedasih, Enzim sekatan, Masalah penjadualan dan penjadualan semula jururawat, Pengurusan penjagaan kesihatan.

Abstract

Nurse shortage, uncertain absenteeism and stress are the constituents of an unhealthy working environment in a hospital. These matters have impact on nurses' social lives and medication errors that threaten patients' safety, which lead to nurse turnover and low quality service. To address some of the issues, utilizing the existing nurses through an effective work schedule is the best alternative. However, there exists a problem of creating undesirable and non-stable nurse schedules for nurses' shift work. Thus, this research attempts to overcome these challenges by integrating components of a nurse scheduling and rescheduling problem which have normally been addressed separately in previous studies. However, when impromptu schedule changes are required and certain numbers of constraints need to be satisfied, there is a lack of flexibility element in most of scheduling and rescheduling approaches. By embedding the element, this gives a potential platform for enhancing the Evolutionary Algorithm (EA) which has been identified as the solution approach. Therefore, to minimize the constraint violations and make little but attentive changes to a postulated schedule during a disruption, an integrated model of EA with Cuckoo Search (CS) is proposed. A concept of restriction enzyme is adapted in the CS. A total of 11 EA model variants were constructed with three new parent selections, two new crossovers, and a crossover-based retrieval operator, that specifically are theoretical contributions. The proposed EA with Discovery Rate Tournament and Cuckoo Search Restriction Enzyme Point Crossover (D_rT_CSREP) model emerges as the most effective in producing 100% feasible schedules with the minimum penalty value. Moreover, all tested disruptions were solved successfully through pre-retrieval and Cuckoo Search Restriction Enzyme Point Retrieval ($CSREP_r$) operators. Consequently, the EA model is able to fulfill nurses' preferences, offer fair on-call delegation, better quality of shift changes for retrieval, and comprehension on the two-way dependency between scheduling and rescheduling by examining the seriousness of disruptions.

Keywords: Hybrid evolutionary algorithm, Cuckoo search, Restriction enzyme, Nurse scheduling and rescheduling problem (NSRP), Healthcare management.

Acknowledgement

First of all, thanks God for the good health and wellbeing that were necessary to complete this thesis. I wish to express my sincere thanks to my parents Lim Soon and Low Siew Pien, and my beloved brothers and sisters. To be loved and trusted by those who really know me best is life's greatest compliment. I could not be able to pursue my doctoral study without their spiritual and financial supports. To the most realistic financial problem, I would like to acknowledge the financial support of MOSTI for the first three and a half years of my studies and Associate Professor Dr. Razamin Ramli for the financial support through FRGS during the last part of my studies. Furthermore, Associate Professor Dr. Razamin Ramli who is my supervisor, a special thank to Dr. for sharing her knowledge, educating me to be strong and wise in facing any challenges, and allowing me the space that I needed to work throughout the journey of my research. I am also grateful to Dr. Xin-She Yang who is from Middlesex University, UK, for the long distance discussions that regards to nature-inspired algorithms, cuckoo search. Also, thanks to Associate Professor Dr. Azizan Saaban, Mrs. Suha, and Mr. Ghassan for the assistances and time contributions to my MATLAB programming. I sincerely thank the supports from the Hospital Sultanah Bahiyah's matron Madam Kamsiah; head nurses of ward ED Sister Shanizam and Sister Azizah Binti Said; head nurse of ward CRW Sister Tan Bee Mee. They spare their time in explaining the ward operational processes and organizing the data for me. Thanks to Penang Adventise Hospital's nurses for sharing their valuable ward experiences as well, especially to senior sister Leong and junior sister Amy. I take this opportunity to express gratitude to all SQS staff for their help and support, especially to Kak Latifah and Kak Aida who assisted me to use SQS's lab for my experimental testing. Besides that, I would also like to extend my gratitude to my supportive roommate Irene, my genuine friends Poy Yen, Carol, Jackkie, Chee Keong, Jimuis, Rosshairry, Meggie Lim, Kak Noor, Kah Wai, Mun Sun, Aunty Lily, Jia Ying, and my fellow colleagues in CAS and COB PhD rooms for being with me throughout the ups and downs. To all, we finally made it, all our sweat and tears finally paid off. Also to my fellow colleagues who are still pursuing their PhD, I believe that all of you will make it as well.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

1PX	One Point Crossover
2F	Two-Factor Blockwise Crossover
2PX	Two Points End Crossover
AB	Agent Based
AC	Adaptive Construction
ACO	Ant Colony Optimization
AI	Artificial Intelligence
ALS	Adaptive Local Search
B	Requested Public Off
B&B	Branch-And-Bound
B&P	Branch-And-Price
BCS	Binary Cuckoo Search
BLP	Binary Linear Programming
C	On-Call's Compensation Off
C1	Order-Based Convention Crossover
CBR	Case-Based Reasoning
CGB	Column Generation Based
CH	Constructive Heuristics
CLP	Constraint Logic Programming
CP	Constraint Programming
CP-CG	Constraint Programming Based Column Generation
CRW	Cardiac Rehabilitation Ward
CSA	Chaos Simulated Annealing
CSREP	Cuckoo Search Restriction Enzyme Point Crossover
CSREP$_r$	Cuckoo Search Restriction Enzyme Point Retrieval
D	Decomposition
DBOP	Day-Based One-Point Crossover
DCS	Discrete Cuckoo Search
DE	Differential Evolution
DNA	Deoxyribonucleic Acid
DP	Dynamic Programming
D$_r$	Discovery Rate Parent Selection
D$_r$_2F	EA with DiscoveryRate Parent Selection and 2FBlockwise Crossover
D$_r$_CSREP	EA with DiscoveryRate Parent Selection and Max[4x4]CSREP Crossover
D$_r$T	Discovery Rate Tournament Parent Selection
D$_r$T_2F	EA with Discovery Rate Tournament Parent Selection and 2FBlockwise Crossover
D$_r$T_CSREP	EA with Discovery Rate Tournament Parent Selection and Max[4x4]CSREP Crossover
DSS	Decision Support System
E	Evening Shift
EA	Evolutionary Algorithm
ED	Emergency Department
EDA	Estimation of Distribution Algorithm
FCA	Forward Checking Algorithm
FLC	Fuzzy Logic Control

FUDS	Fitness Uniform Deletion
FUSS	Fitness Uniform Selection Strategy
GA	Genetic Algorithm
GC	Graph Colouring
GCS	Guided Complete Search
GL	Adaptive Heuristics-Greedy Local Search
GP	Goal Programming
GPU	Graphics Processing Unit
GRASP	Greedy Random Adaptive Search Procedure
H	Heuristics
HB	Hybridization
HGHCA	Hybrid Genetic and Hill Climbing Algorithm
HH	Hyper-Heuristics
HO	Heuristics Ordering
HSB	Hospital Sultanah Bahiyah
ICS	Improved Cuckoo Search
IGA	Indirect Genetic Algorithm
IHCS	Improved Hybrid Cuckoo Search
ILP	Integer Linear Programing
IP	Integer Programming
KB	Knowledge-Based
L	Unexpected Leave
L	L-Shaped Method
LP	Linear Programming
LS	Local Search
M	Morning Shift
MA	Memetic Algorithm
MH	Meta-Heuristics
MIP	Mixed-Integer Linear Programming
MM	Maximax and Maximin Parent Selection
MM_2F	EA with Maximax and Maximin Parent Selection and 2FBlockwise Crossover
MM_CSREP	EA with Maximax and Maximin Parent Selection and Max[4x4]CSREP Crossover
MO	Mathematical Optimization
MOEA	Multi-Objective Evolutionary Algorithm
MP	Mathematical Programming
N	Night Shift
NBOP	Nurse-Based One-Point Crossover
NBRS	Randomly Selected Nurse-Based Crossover
NHS	National Health Service
NN	Binary Neural Networks
NRP	Nurse Rescheduling Problem
NSP	Nurse Scheduling Problem
NSRP	Nurse Scheduling and Rescheduling Problem
O	Night Shift's Compensation Off
OBX	Order-Based Crossover
OR	Operations Research
P	Public Off
PMX	Partially Mapped Crossover

PSO	Particle Swarm Optimization
PUX	Parameterised Uniform Order Crossover
Q	Requested Weekend Off
R	Request Off
REP	Restriction Enzyme Point
RCPSP/τ	Resource-Constrained Project Scheduling Problem
RGA	Real-Valued Genetic Algorithm
Rk	Rank-based Parent Selection
Rk_2F	EA with Rank-based Parent Selection and 2FBlockwise Crossover
Rk_Row	EA with Rank-based Parent Selection and Row-wise Crossover
ROW	Row-wise Crossover
RPBIL-DE	Hybridization of Real Code Population-Based Incremental Learning with Differential Evolution
SA	Simulated Annealing
SS	Scatter Search
T	Requested Weekly Off
T	Tournament Parent Selection
T_2F	EA with Tournament Parent Selection and 2FBlockwise Crossover
T_CSREP	EA with Tournament Parent Selection and Max[4x4]CSREP Crossover
T_Row	EA with Tournament Parent Selection and Row-wise Crossover
TS	Tabu Search
U	Weekly Off
UOX	Uniform Order Based Crossover
VNS	Variable Neighbourhood Search
W	Weekend Off

CHAPTER ONE

INTRODUCTION

Nursing is a noble and respectable profession. It is a profession that is acknowledged worldwide due to its significant contribution to the societal well-being. However, this profession brings considerable challenges that affect the nurses' performance. Hence, a scientific investigation to improve their performance is timely.

1.1 Background of the Study: Nurses' Working Environment

Nurse scheduling and schedule readjustment in real time operation has become a challenging global issue. This calls for a critical attention as nurses generally make up approximately 60% of employees in hospitals worldwide (Chiaramonte, 2008; Eastaugh, 2007), which also denotes a sizable amount of financial allocation (Dunton, Gajewski, Klaus & Pierson, 2007). In Malaysia, the Ministry of Health received a national budget of 7.66% in 2013 (Ministry of Health Malaysia, 2013). However, despite such allocation, a survey carried out by the Ministry for Healthcare Services during the year 2006-2008 found that government hospitals as opposed to their private counterparts incurred the highest number of complaints (Ministry of Health Malaysia, 2008). The survey revealed an increment of more than 67% (i.e. 150-250 cases) during this period. The level of complaints received clearly signifies the unsatisfactory quality of the nursing services, which can affect patient safety. As Tang and Ghani (2012) reported that working conditions was one of the main factor to determine job satisfaction. The situation is worsened when there is a shortage of nurses (Ministry of Health Malaysia, 2004; Missouri State Board of Nursing, 2008).

In addition to the nursing shortage, there are other work-related aspects that affect the unsatisfactory quality of the nursing services or inferior engagement of nurses at work. They are nurse demands, absenteeism, burnout and pressure. These aspects are discussed as follows.

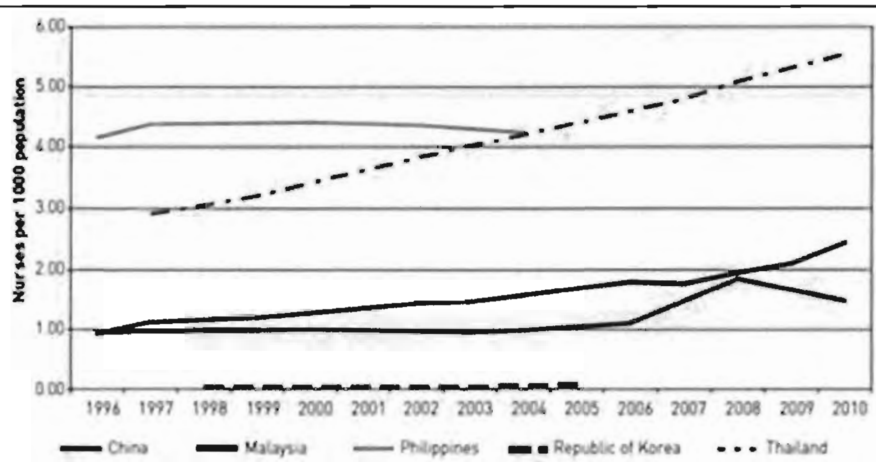
1.1.1 Nurse Demand

> "*The whole care system is currently facing the huge challenge of delivering care at a time of increased demand and scant staff resource. This is the reality which nurses face in every working day.*" *Peter Cater,* Chief Executive of Royal College of Nursing (Ford, 2013)

Nurses displeased with overloaded demands of over 8-hour shift length was reported (Ingersoll, Olsan, Drewcates, DeVinney, & Davies, 2002). The stress had to certain extent intensified the likelihood of leaving among nurses (Hayes et al., 2006; Tourangeau & Cranley, 2006), which in turn leads to a crisis in nurse shortages (Judith, 2006).

Malaysia is no exception to such problem. Indeed, many nurses had left their profession or retired each year (Hayes et al., 2006). Over 70% of Malaysian hospitals were facing insufficient nursing staff, therefore, nursing programs were increased to elevate the nursing profession to professional status (Bernama, 2011; Ministry of Health Malaysia, 2012). In fact, the insufficiency was still involved with a great number of nurses. Statistically, government hospitals obtained 50,063 nurses and private hospitals obtained 24,725 nurses, as Ministry of Health Malaysia (2012) reported. In addition, many trained nurses had gone to foreign countries for more attractive working environment and remuneration (Ministry of Health Malaysia, 2004; Missouri State Board of Nursing, 2008). Currently, approximately 20% of Malaysia's 84,000 nurses are estimated to be working overseas (Riot, 2012). For example, many skilled nurses from Malaysia were recruited by hospitals in Singapore to fill the shortage there (Barnett, Namasivayam, & Narudin, 2010).

Nurse turnover is a widely acknowledged problem worldwide. Such phenomenon is also observed among nurses in the western pacific countries such as China, Malaysia, Philippines, Republic of Korea, and Thailand (Malaysia Health System Review, 2013). Figure 1.1 shows the ratio of nurses per 1000 population in the period of 1996-2010 provided by the WHO Regional Office (Malaysia Health System Review, 2013). As shown, besides a slight improvement of nurse-to-population ratio in the Republic of Korea (from 3:1000 [1997] to 5.6:1000 [2010]), other countries, such as China and Thailand, still witnessed a more severe nurse shortage.



Source: WHO Regional Office for the Western Pacific, Health indicators database, and Thailand Health Profile reports

Figure 1.1. Ratio of nurse-to-population

In Malaysia, the ratio of nurse-to-population was slightly better during a ten-year period between 1996 and 2006. Table 1.1 shows a noticeable increase of 27586 nurses during this period (Malaysia Ministry of Health, 2007). The statistics showed some improvement in the nurse-to-population ratio from 1:1055 in 1996 to 1:559 in 2006.

Table 1.1

Malaysia Nurse Employment in Year 1996 to 2006 from Ministry of Health (2007)

Year	*Total no. of nurses (Jururawat Desa)*	*Nurse : population*
1996	20 056	1:1055
1997	24 545	1:883
1998	23 672	1:937
1999	27 236	1:834
2000	31 129	1:747
2001	33 295	1:715
2002	35 280	1:695
2003	36 784	1:681
2004	40 220	1:636
2005	44 120 (15 618)	1:592
2006	47 642 (16 667)	1:559

Source: Ministry of Health, Malaysia (2007).

In another survey by the Ministry of Health Malaysia (2013), the nurse-to-population ratio improved to 1:345 in 2012 compared to the 1:410 ratio in 2010 (Health Indicators, 2010). Though the improvement suggests that shortage crisis over the years is being addressed, the ratio is still lagging behind the World Health Organization's standard ratio of 1:200 (Ministry of Health Malaysia, 2013).

Despite the shortage, turning to external aids such as recruiting a new cadre of nurses, temporarily enlisting help from nurse agency services, and obtaining overseas expertise are not desirable because these measures can cause long-term problems (Barnett, Namasivayam, & Narudin, 2010), as they may lead to unqualified nurse issue (Barnett et al., 2010) and increase hospitals' operation cost due to expensive recruitment (Maenhout & Vanhoucke, 2011).

Given the difficulty to attain a balanced nurse demand-supply, optimizing the use of the existing nursing resources is the best alternative but requires effective nurse scheduling.

1.1.2 Nurse Absenteeism

Absenteeism is defined as a state of being unavailable for a job, at a particular time (Clark & Walker, 2011). Incidence whereby an employee who is pre-planned to carry out a specific task but is not available due to unplanned absences or staff turnover, is considered a disruption in a personnel schedule (Maenhout & Vanhoucke, 2011). Some of these instances are when leaves are taken due to personal obligations to accommodate sick family (McMenamin, 2010), unexpected own illness (Maenhout & Vanhoucke, 2011), maternity leave (Maenhout & Vanhoucke, 2011), unexpected outpatient cares, training programs and urgent meetings (Barnett, Namasivayam, & Narudin, 2010), all of which forcing nurses to leave their duty.

Beside these, another factor that disrupts the nursing schedule is the issue of absenteeism amongst nurses (Moz & Pato, 2007) which greatly jeopardizes staffing budget of hospitals (Maenhout & Vanhoucke, 2011). The study of Maenhout and

Vanhoucke (2011) pointed out that schedule disruptions due to absenteeism has incurred a budget loss of 4% off the total resources allocated for staffing. In another instance, a survey by the Press Association (2011) reported budget waste of approximately £290 million in National Health Service (NHS) due to junior nurse sickness absence crisis.

Besides nurse understaffing, taking of sick and emergency leave amongst nurses also affect nurses' performance (Al-Ahmadi, 2009). The out-of-normal counts of sick leaves by nurses also reflect poor health among nurse (Barnett et al., 2010; Ford, 2013; The Press Association, 2013).

Not only absenteeism gives rise to nurse understaffing issue, it further causes schedule sustainability problem (Moz & Pato, 2007; Bard & Purnomo, 2005). Yet, initiating considerably frequent change to address schedule disruption due to uncertain events can be problematic in nurse management. This view is evident in past studies which revealed the preference of minimizing retrieval change among hospitals (Moz & Pato, 2007; Maenhout & Vanhoucke, 2011). Further, it is also noted that making radical change to a postulated schedule may be less problematic compared to addressing frequent imminent changes (Clark & Walker, 2011). According to Gutjahr and Rauner (2007), it is vital to improve the current-day decisions from a perspective of long-run advantages. Therefore, any attempt to improve the nurse scheduling should be swift enough to cater for the unexpected change, particularly those of the imminent.

1.1.3 Nurse Fatigue

Work-related fatigue or burnout is defined as an emotional exhaustion that shows lack of personal accomplishment (Vahey, Aiken, Sloane, Clarke, & Vargas, 2004). There are several reasons how fatigue amongst nurses is associated with the understaffing climate in hospital. Kalisch and Aebersold (2010) revealed that every nurse was bound to at least 34% of highly complex multi-tasking work time. Besides, unexpected on-call duty (Banyal, 2011) and irregular or unpredictable working hours such as night and weekend works also give rise to stress and burnout among nurses (Horrocks & Pounder, 2006; Landro, 2008). There were also times where nurses were forced to work between wards and even worked to the extent of sacrificing their break (Ford, 2013; Tang, Sheu, Yu, Wei, & Chen, 2007). Frequent disruption in the nurse work schedule (e.g., increase in the nurses' workload) and long shifts basis (e.g. ten hours shift and above) can lead to nurse burnout (Cohen & Golan, 2007; Shahriari, Shamali, & Yazdannik, A, 2014; Stimpfel, Sloane, & Aiken, 2102).

The discussion above clearly demonstrates that fatigue could be one of the key factors that jeopardize nurses' ability and willingness to be engaged in their job. As pointed out by Gormley (2011), nurse burnout is closely related to low level of nurse engagement. In fact, low engagement has also been noted to affect satisfaction in patient care (Aiken, Clarke, Sloane, Sochalski, & Silber, 2002; Needleman et al., 2011). These altogether would mean poor nursing services. Poor service, according to Tang et al. (2007), include such incompetence as long waiting time, low consciousness of patient hygiene, uncomfortable ward environment, and impatient attitudes amongst nurse.

Studies have also reported that nurse fatigue led to the increase of work-related injuries risk (Querstret & Cropley, 2011), high patient mortality, morbidity and adverse event rates (Dunton et al, 2007; Needleman et al, 2011), as well as medication errors (Johari, Shamsuddin, Idris, & Hussin, 2013). The more severe impact of fatigue could be imagined in that it risks patient's life during critical periods where attentive patient care is needed.

Specifically, Tang et al.'s (2007) study pointed out that heavy workload, personal neglect, and fresh nurse as three prominent reasons of medication error. According to Kalisch and Aebersold (2010), an error prone of 1.5 errors per hour was reported for hospitals. Such indication has critical implication for hospitals in Malaysia. As highlighted by Johari et al. (2013), there were 2572 adverse cases reported in 2009. The Malaysian hospital was portrayed as having a high probability of making medical error.

Building on the above points, a practically effective scheduling model should therefore seriously consider the effect of fatigue amongst nurses so that their competency would not be hampered.

1.1.4 Nurse Personal Pressure

Nurse pressure affects adversely nurse engagement. The Point of Care Foundation (2014) reported only 27% of nurses surveyed were actively engaged in their nursing work. The Nursing Board of Malaysia has long emphasized caring and teamwork, respect for human dignity, and community participation to reduce the pressure nurses face at work but to little avail (Ministry of Health Malaysia, 2008). Majority of nurses felt unappreciated for their voice was not heard (Ford, 2013, September; Gormley, 2011), hence demolishing their expectations (Stephenson, 2014; Valouxis et al., 2012). They also reported to being treated unfairly (Kane-Urrabazo, 2006) and did not engage in a decision-making process (Bard & Purnomo, 2005). These all contribute to work pressure resulting in burnout and depression that could jeopardize their marital life. Indeed, this occupation often sees a high divorce rate at 28% (McCoy & Aamodt, 2010).

Without the support from the head nurse, nurses may find it difficult to balance their work schedule and family commitments (McEachen & Keogh, 2007). This difficulty is exacerbated by the timely-off duty's preferences such as compensation off duty, requested off day, weekend off duty, and public holiday, an issue that lacks scientific investigation (Chiaramonte, 2008). Given that many nurses prefer their weekly off is scheduled providentially on some particular days such as weekend or other personal significant days like anniversary, birthday etc. (Chiaramonte, 2008). Though this is highly preferred, the chances of getting timely off on these days are small, which may lead to divorce and depression. Compensation pay rather than off day as a reward to those who had been on consecutive night shifts or on-call duty may exacerbate further the physically and mentally depressed. This is because the night duty requires that nurses remain awake (Horrocks & Pounder, 2006) and remain alert to on-call duty though they are not scheduled to work on that day (McEachen & Keogh, 2007), when they are supposed to be resting.

Besides that, nurse gets unmotivated when they have to endure internal enforcement while undertaking busy workload (Clark & Walker, 2011; Maziah, Wichaikhum, &

Nantsupawat, 2012). There is a relational aggression to the generational changes of old nurses with new nurses. For instance, over reliance on a senior nurse by a newly graduated nurse (Barnett et al., 2010) inevitably leads to internal strife such as horizontal or vertical bully (Dellasega, 2009) where the same nurse may end up working often to an on-call duty or late night duty. When the unfair delegation happens, respect for the head nurse is lost (Banyal, 2011; Clark & Walker, 2011; Kane-Urrabazo, 2006). Perhaps, many head nurses today are rather seeking for a quick-fix solution with little consideration to the long-term consequences when they make decision about nurse allocation (Kane-Urrabazo, 2006). These aggravations of bureaucracy conceivably enforce a less harmonious working environment, which affect the personal lives of nurses (Barnett et al., 2010).

When the nurses feel that they are not treated fairly and do not receive the necessary support from the head nurse, pressures build up, leading to medication errors (Tang, Sheu, Yu, Wei & Chen, 2007) and high turnover (Flynn, Mathis, & Jackson, 2007; Gormley, 2011; Hayes et al., 2006; Heinrich, 2001; Tourangeau & Cranley, 2006), which in turn worsen the already critical shortage of nurses (Aiken, Clark, Sloane, Sochalski, & Silber, 2002; Buchan & Calman, 2004; Cohen & Golan, 2007, Eastaugh, 2007; Heinrich, 2001).

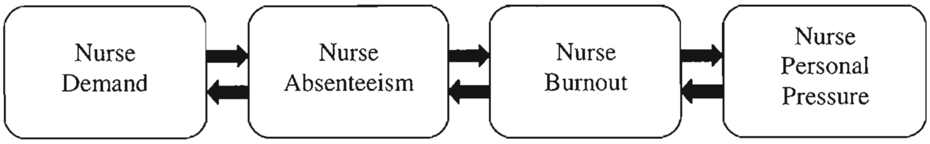


Figure 1.2. Unhealthy working environment for nurses

In sum, the inter-connectedness of the four major conditions discussed above creates an unhealthy working environment for nurse (refer to Figure 1.2), which pressures on nurses' quality service. To address the issue of low quality in nursing services, an effective schedule of staffing is suggested. This study thus aims to fill this void by developing a nurse scheduling and rescheduling model that is able to produce desirable and re-adjustable schedule in real time, as illustrated in Figure 1.3. Having pinpointed the impact of uncertainty, nurse capacity, and nurse preference earlier, this model tackle the nurse scheduling and rescheduling problem (NSRP) by considering the factors.

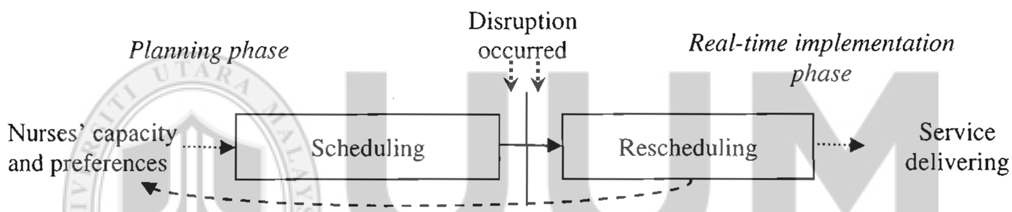


Figure 1.3. A process of nurse scheduling and rescheduling

1.2 Nurse Scheduling and Rescheduling Approaches

A nurse scheduling problem (NSP) is a matter of allocating a number of nurses over a time period to perform a set of shift duties (Gutjahra & Rauner, 2007). But, uncertain retrieval conditions occur in all nurse scheduling problems, called a nurse rescheduling problem (NRP). Over the years, several approaches or techniques have been explored to address these problems. Such approaches can be categorized into four: (1) mathematical optimization in NSP (Azaiea & Al Sharif, 2005; Glass & Knight, 2010) and NRP (Bard & Purnomo, 2005b; Clark & Walker, 2011); (2) heuristics in NSP (Brucker, Burke, Curtois, Qu, & Berghe, 2010; Horio, 2005) and NRP (Baumelt, Dvorak, Sucha, & Hanzalek, 2013); (3) meta-heuristics in NSP

(Aickelin & Dowsland, 2004; Baumelt, Sucha, & Hanzalek, 2007; Suman & Kumar, 2006) and NRP (Maenhout & Vanhoucke, 2011); and (4) knowledge-based in NSP (Akihiro, Chika, & Hiromitsu, 2005; Kumara & Perera, 2011) and NRP (Beddoe, Petrovic, & Berghe, 2002). However, one limitation of these approaches is that because they focus on a specific technique alone they are not able to solve complex requirements of nurse's real personal preferences. For NSP and NRP, nurse preferences are always the offset point whenever nurse coverage is challenged by demand fluctuation or incompetent scheduling (Chiaramonte, 2008; Punnakitikashem, 2007). Although nurse preferences have been less considered, the hybridization of the four approaches to some extent would tackle complex nurse requirements and preferences (Goodman, Dowsland, & Thompson, 2009; He & Qu, 2012; Winstanley, 2004). This situation implies that a hybridization approach could provide a better solution for the NSRP.

A nurse scheduling problem is often complicated by large numbers of constraints and required to be solved quickly in a polynomial time, which is a NP-hard problem (Cheang, Li, Lim, & Rodrigues, 2003). However, there is no single approach that can possibly fully solve such problem quickly when a big problem is involved (Horio, 2005; Winstanley, 2004). In fact, the amount to a search space for NSP can be counted by S^{n*d}, where S denotes the types of shifts, n denotes the total of nurse, and d denotes the scheduling period (Winstanley, 2004). As such, this is a big enumeration where $13^{(39*14)}=13^{546}$ is formed by a fortnight schedule with 39 nurses and 13 shift types. But, the amount is reduced slightly by decreasing the domain size. For that reason, the static mathematical optimization based techniques and knowledge based techniques alone may be not suitable to be used, whereas approximation algorithm rooted in flexible attribute has more potential for hybridization, such as, the Evolutionary Algorithm (EA).

Moreover, in order to minimize retrieval change in NRP (Moz & Pato, 2007; Punnakitikashem, 2007), some further considerations are unnoticed by previous NRP techniques. For instance, there is lack of sensitivity for giving suitable recommendation because past and current shift attentions are considered but not future attention (Clark & Walker, 2011; Punnakitikashem, 2007). Also, rescheduling by Clark and Walker (2011) takes the original schedule into account but not vice-versa. All of these show that the short-sided retrieval approaches are suitable to address the disruption that occurs on that day (e.g. current impromptu condition) but do not take into account the long term consideration. By considering NSP and NRP in this scheduling problem, a schedule that considers both weaknesses can be developed.

Technically, a rescheduling phase has a smaller search space than a scheduling phase (Clark & Walker, 2011), so it is supposed to be an effortless search in rescheduling. However, with some additional restrictions, the difficulty of searching an optimal one might not be lesser than that in the scheduling phase. For that reason, an approach that is able to make changes while sustaining the restricted space is needed. Indeed, Cohen, Stuenkel, and Nguyen (2009) noted the lack of flexibility in scheduling and rescheduling approaches in a condition that requires frequent changes and a relatively big number of constraints that need to be satisfied. This gives EA a potential approach to employ because its flexibility can enhance the exploration and exploitation search.

Essentially, EA's strength lies in searching for a variety ways with a degree of sophistication. It produces more than one solution at a time and then obtains a better one which predominantly serves the dilemma point of pursuing schedule stability but changes due to disruption. However, EA challenge has never ended because slow convergence could appear and even affect an output's feasibility while exploration is dominant. In contrast, a severe loss of population diversity situation leading to premature convergence shall occur while exploitation is dominant with respect to exploration. To balance both, these conditions have led to more technical challenges to the modus operandi of EA and enhanced EA's operators (e.g., initialization, parent selection, crossover, mutation, and regeneration) caused by hybridization. Essentially, for a more flexible search we accept uncertain individuals' attribute in a population. Therefore, by balancing between exploration and exploitation, a flexible search can give further insight into our new selection operator and recombination operator.

1.3 Problem Statement

There are issues of nurses leaving the service due to undesirable work schedule (Aiken et al., 2002; Chiaramonte, 2008). The unsatisfactory schedule is worsened when the nurses are absent on the day, prompting the head nurses to rectify the impromptu schedule. Such problem will never end since the well-being of nurses and ward operating requirements are vital elements that need to be taken into account concomitantly, even in such situation. Therefore, generating a more flexible schedule system that involves scheduling and rescheduling is valuable for the head nurse. Literatures indicate a substantial work of nurse scheduling or rescheduling has been conducted but the integration of nurse scheduling and rescheduling is very limited. To the researcher's knowledge, Chiaramonte (2008) was the only one who carried out a research that had considered the integration of nurse scheduling and rescheduling problem (NSRP), but overlooked nurse time preferences (Chipas & McKenna, 2011), fair on-call delegation (Clark & Walker, 2011), quality and quantity change of retrieval (Clark & Walker, 2011), and lack of examination on how scheduling and rescheduling are interrelated in determining the seriousness of disruption (Maenhout & Vanhoucke, 2013). Even though these elements were recommended to be considered, they have never been actually executed. Hence, our research aims to incorporate both the scheduling and rescheduling aspects, which additionally require flexibility of search.

Nevertheless, the randomness strength of evolutionary algorithm (EA) is in excess of exploration (Grosan & Abraham, 2007). Though EA is capable to search larger search space but it has less effective in identifying local optima in term of computational time and the quality of solutions (Aickelin & Dowsland, 2004). Perhaps, cuckoo search algorithm (CS) that obtains a good balance of intensive local search strategy (Yang & Deb, 2010) may well fit to the limitation of EA. To our knowledge, the combination of genetic algorithm and cuckoo search is very limited. It has only been used to solve runway dependent aircraft landing problem (Zheng, Zhou, & Guo, 2013), job shop scheduling problem (Abu-Srhahn & Al-Hasan, 2015; Singh, Kurmi, & Tiwari, 2015), and traveling salesman problem (Ala'a Abu-Srhan & Al Daoud, 2013). However, there are no studies that utilized the hybrid evolutionary algorithm with cuckoo search algorithm on nurse scheduling problem, even to any personnel scheduling problems. In the only one work of NSRP, Chiaramonte (2008) employed the agent based technique. Therefore a hybridization of EA and CS has motivated this research.

In order to solve this complex NSRP, an enhancement of flexibility attribute in evolutionary algorithm (EA) is urged. EA consists of multi-stage operations which are initialization, parent selection, crossover, mutation, and replacement. In order to introduce the flexibility attribute in EA, potential improvements in parent selection operator is still possible with regards to binary tournament parent selection of Sharma and Mehta (2013) from the elite perspective. This elite act focuses mostly on exploitation in parent selection. On the flipside, the selection pressure of dissimilarity relationship (Yang & Deb, 2010) between parents in an uncertain population circumstance (Heizer & Render, 2006) could be introduced as part of a partner selection strategy. This could be another substance to increase exploration to release it from the control of population diversity, as studied by Lim and Ramli (2014).

In crossover operator, two improvements of permutation by different types of cross-points in terms of size division and direction can be targeted to further enhance the crossovers with nurse-based cross-point of Moz and Pato (2007) and row-wise crossover of Ramli (2004). These two strategies considered as fixed horizontal cross-point behavior could limit the permutation as well as exploration, though their large size of crossing division is promising to reduce constraints disruptions. Moreover, as for repairing purpose in our NSRP, the partial concept of cuckoo search is possible to enhance the Maenhout and Vanhoucke's (2008a, 2011) crossover that regards the deliberate cross-point. As a result of their approach, fast convergence was obtained due to lack of exploration by crossing only to case-specific constraint violations, though this specific crossover is able to remain a good sub-solution. For these reasons, an integration of vertical and horizontal cross-points as well as a direct type of crossing point with flexible crossing sizes are studied to balance exploration and exploration in crossover operator. Thus, investigating a higher degree of permutation for crossing over is the aim of our research.

1.4 Research Questions

This study is endeavored to answer the following questions:

1. What constraints are considered necessary in the scheduling and rescheduling model?
2. What is the methodology needed to generate an efficient schedule?
3. How does nurse rescheduling system adjust nurses' shift to solve daily fluctuations?
4. What operators of the selected methodology shall be improved?
5. How do we evaluate the effectiveness of the proposed models?

1.5 Research Objectives

Essentially, the aims of this research are to develop a nurse scheduling and rescheduling model by looking into a hybridization of evolutionary algorithm and cuckoo search technique for solving real-time scheduling instances. In essence, the final outcome is a priory prepared *best-so-far* nurse schedule that involves impromptu decision making to produce a feasible nurse schedule, in case of rescheduling. Minimizing constraint violation is targeted to solve a particular uncertain nurse scheduling and rescheduling problem (NSRP).

The enhanced technique of evolutionary algorithm and cuckoo search for NSRP is identified and used to accomplish the specific objectives as below:

1. To identify all relevant constraints and parameters that make up all hard rules and nurses preference as far as possible within appropriate nurse skills and staffing size.
2. To determine the change of adjustment that gives a low impact on other nurses in a rescheduling problem
3. To construct newly modified parent selection operators to acclimatize population diversification.
4. To construct new modified crossover operators for a scheduling problem and present it as a repair operator for the rescheduling problem. They are to promote a more flexible way of crossing over and enhance the exploitation element to evade slow convergence.
5. To evaluate the performance of several evolutionary models and the proposed nurse scheduling and rescheduling model with *what-if* analysis

In conclusion, the hybrid EA reinforces the desire for the equilibrium between exploration and exploitation in order to search for the best-so-far solution that satisfies the identified hard, semi-hard and soft constraints in nurse a scheduling and rescheduling problem.

1.6 Research Contributions

Basically there are two main contributions of this research. They are theoretical contributions of Evolutionary Algorithm-based hybridization and practical contribution to a nurse management system.

Firstly, this research constructed a new hybrid evolutionary algorithm and a cuckoo search approach by looking into parent selection and crossover operators. Three enhanced parent selection operators for manipulating population diversity are named as below:

a) Maximax and Maximin parent selection
b) Discovery Rate parent selection
c) Discovery Rate Tournament parent selection

Two modified matrix crossover operators that achieved a more flexible way of crossing over are named as below:

a) Two-factor Blockwise crossover
b) Cuckoo Search Restriction Enzyme Point crossover

One noticeable point is that the cuckoo search-based crossover operator can be applied to a rescheduling problem due to an integration of Restriction Enzyme Point strategy. Hence, the concept of restriction enzyme in DNA has first been practically implimented in crossover. In all, these are to enrich the exploitation element to evade slow convergence for the sake of over flexibility. Besides that, an enhanced fitness calculation approach with regards to hard, semi-hard, and soft constraint strategy is applied in this research. It is suitable for organizing and handling the priority of a vast number of constraints as well as the complexity level of a constraint. The semi-hard constraint strategy is newly introduced to link between the scheduling and rescheduling.

As the second main contributions, this research provided an advancement of computerizing the combination of nurse scheduling and rescheduling to improve nurse productivity and services. The final tested model is able to produce desirable and adjustable schedules that utilize the available nurses without neglecting their preferences. In particular, the integration of nurse scheduling and rescheduling problem (NSRP) focused on solving timely nurse preferences, fair on-call delegation, quality and quantity change of retrieval, and two-way dependency between scheduling and rescheduling by examining the seriousness of disruption.

1.7 Scope of the Research

The present research is about a shift assignment which integrates nurse scheduling and nurse rescheduling into a model for a ward in a hospital environment. Nurse task assignment is, however, not within the scope of our research because it is the final stage of nurse management system (see Table 2.1). Furthermore, nurse task assignment involves nursing care knowledge and head nurse decision after the doctor has diagnosed a patient (e.g., clinical knowledge on medication treatments and injections, patient condition).

The final model can be operated by any ward user (head nurse) of any hospital. For the purpose of illustrating the application of scheduling and rescheduling model in a critical ward such as an Emergency Department, a Cardiac Rehabilitation Ward was also chosen since cardiothoracic needs require more complex care and nurse's commitment. As 25.4% diseases of the circulatory system are the number one cause of death in hospitals, approximately 63% of that mortality rate was cardiac patients by year 2010, according to Ministry of Health (Health Indicators, 2010).

Matrons, head nurses and staff nurses were interviewed to gather data on some real elements to be incorporated into the model. For example, data were collected nurses' availability, their work environment in an ordinary period or uncertain period, their preferences and specific needs, and skill requirements of a particular ward.

1.8 Definition of Key Terms

Below are the key terms that used in this research:

- *Artificial intelligence* is a field of computer science, which is study on the automation of intelligent behavior (making a computer reason in a manner similar to humans) regarding perception, reasoning, action and computation.
- *Continued service* is delivering 24 hours, 7 days a week service.
- *Coverage* is a number of nurses who are taking ON duty in a ward.
- *Disruption of schedule* is changes upon schedule in an occurrence when a nurse is unavailable to work in a shift as scheduled due to unplanned absences or other nurse turnover.
- *Feasible solution* or *feasible schedule* is an acceptable output that satisfies all hard constraints which apply to all nurses.
- *Integrated Request Off* concept is the off day which is authorized by a head nurse to complement the off day with off day requested by a staff nurse.
- *Mutual respect* is pursuing a harmonious working condition by searching a tolerable point between a head nurse and staff nurses in shift allocation, especially in a particular day off duty.
- *Nurse assignment* is assigning nurses to individual shifts based upon caseload method that ensures each nurse is charged with an equal number of patients with certain nursing skill required.
- *Nurse management system* is the four stages of managing the nursing personnel. The stages nurse planning, scheduling, rescheduling, and assignment.

- *Operations research* is a scientific approach to decision making sought on a quantitative and rational basis, usually dealing with the allocation of scarce resources.
- *Planning period or horizon* is the long-term temporal period for which personnel staffing is done, e.g., determining and scheduling the number of personnel needed during a year.
- *Premature convergence* is where population soon loses its diversity resulting in a search confined into a suboptimal solution that causes an optimizer to be stuck.
- *Quality of schedule* has a minimum fitness achievement corresponding to the satisfaction of ward coverage and nurses' preference.
- *Reactive scheduling, rescheduling* and *real-time scheduling* are synonymous. They refer to a process of revising the set of scheduled nurses for a shift that signifies the daily/hourly adjustment of nurses.
- *Real-time* is a time of implementing or executing a planned schedule. It strongly corresponds to changes.
- *Schedule* or *roster* is a set of work shift, off day, and tasks arrangements where a personnel is assigned to a particular shift in a particular day.
- *Scheduling* is a process of creating a personnel schedule. It is a subset of a planning period and normally scheduled on a weekly, fortnightly or monthly basis.
- *Slow convergence* is a situation where an algorithm spends time exploring uninteresting regions of search space.
- *Unfair nurse on call delegation* is a situation where some nurses get no changes but others are required to change the number of shifts by a head nurse during rescheduling.

- *Work shift* is a period of time within a day for which a nurse will perform work.
- *Work stretch* is a continuous chain of work shifts.

1.9 Thesis Organization

This research started by showing a backdrop of the present nursing situation, which revealed the gap that needed to be filled.

Chapter Two provides an in-depth understanding of the present nurse scheduling and rescheduling problem. It also reviews some techniques that have been implemented to address the nurse scheduling and rescheduling problems.

Chapter Three reviews related literatures on search techniques. This chapter highlights some strengths and weaknesses of the present EA and CS pertaining to their fundamental principles, hinting how the techniques can be improved.

Chapter Four describes the overall construction of the models as well as the nurse scheduling and rescheduling prototype. The evaluation strategy to validate the models is shown in the next chapter.

Chapter Five discusses the implementation of the models with several experimental testing. By then, some objectives of the research have been achieved.

The last chapter, Chapter Six, gives an overall conclusion based on the findings. Moreover, limitations of the present study and recommendations for future research are highlighted.

CHAPTER TWO

REVIEW OF NURSE SCHEDULING AND RESCHEDULING PROBLEMS

Developing a nurse schedule is a complicated task that requires consideration of staff nurses and head nurse. This chapter starts by introducing the fundamental problem of scheduling and rescheduling, following which the need for an amalgamation of nurse scheduling and rescheduling is discussed. The chapter also sheds light on the factors a head nurse should take into consideration whilst developing, formulating and implementing a nurse schedule. In addition, the objective function and constraint which formed by three main factors (i.e., nurse capacity, preferences and uncertainty) are also reviewed. Subsequently, nurse scheduling and rescheduling techniques that have been applied are reviewed.

2.1 Nurse Management System

At planning stage, a good planner focuses on identifying what, when, how and who to assign a particular task whilst keeping the available resources in hand for its responsive completion (Billings, 1985). Next, scheduling intends to match those identified elements of what, when, how and who to achieve effective job completion. Bradley and Martin (1990) asserted that in a nurse management system, three inter-related nursing human resource decision phases are involved. They are nurse planning, nurse scheduling, and nurse allocation. However, some scholars suggested that nurse management system could be specifically divided into four stages which are budgeting, scheduling, rescheduling, and task assignment (Azaiez & Al Sharif, 2005; Punnakitikashem, 2007; Punnakitikashem, Rosenberger, Behan, Baker, &

Goss, 2006; Chiaramonte, 2008). In the last stage, nurse rescheduling and task assignment are the separations from nurse allocation. In table 2.1, time horizon attribute provides a clear understanding for classifying the various stages of nurse management system. In each stage, decision is made within its time horizon to address certain types of areas, problems or concerns. A summary of the nurse management classification is shown in Table 2.1.

Table 2.1

Classification of Nurse Management System

3 stages of nurse management system	4 stages of nurse management system	Time Horizon	Types/Examples
1. Staffing process/nurse planning	1. Nurse budgeting/ Staffing decisions	Long term (annually)	Making plans and decisions to meet available budgets; predicting patients` care, monitoring regulations by nursing union and forecasting nurse hiring.
2. Nurse scheduling	2. Nurse scheduling	Mid term (several weeks)	Identifying minimum number of nurses required per shift; Developing schedules; specifying availability and unavailability of nurses and assigning duties to nurses whilst ensuring peak and off peak operating hours.
3. Nurse allocation	3. Nurse rescheduling	Short term (1-3 days or hours)	Recruiting additional nurses to facilitate in absenteeism and patient classification systems
	4. Nurse task assignment	Short term (30 minutes)	Workload balance consideration, intuitive judgment or caseload method

At the start, 'Staffing Process' is the first level of nurse management system. It is also known as nurse planning, staffing decisions, or nurse budgeting. This stage primarily focuses on forecasting to determine the total number of nurses needed during a year in order to finalize the required budget. The second stage is 'Nurse Scheduling', which deals with formulating schedule of nurses based on the predicted number of patients whilst keeping the number of available nurses beforehand. Lastly is 'Nurse Allocation' which includes nurse rescheduling and nurse task assignment and deals with allocation and management of personnel to actual work sites.

The difference between the nurse rescheduling and nurse task assignment is that the former revises a set of nurses scheduled for a shift within approximately 90 minutes (Punnakitikashem, 2007; Punnakitikashem et al., 2006) before each shift. The nurse task assignment decision, however, assigns nurses to individual shifts based on the caseload method (Azaiez & Al Sharif, 2005). This ensures that each nurse is allocated the same number of patients. However, the nurse task assignment is not explored in this research since it involves taking the complicated and unpredictable individual patient's health condition into consideration. Therefore, the literature review discusses the first three levels of nurse management system which are planning, scheduling and rescheduling.

2.1.1 Planning in the Nurse Management System

Planning is one of the basic functions of management (Billings, 1985). McEachen and Keogh (2007) and Eastaugh (2007) noted that the core aim of planning is to provide higher output with lesser input and efforts. This clearly outlines the core vitality of nurse planning and its significance in balancing all factors such as managing the number of available nurse per shift in a ward whilst providing an optimal level of quality care beforehand.

Besides the size of ward or department, good planning and allocation of skilled resources are crucial tasks to be performed by the nurse manager. In the past, Billings (1985) used traditional management terms to describe nursing administration which including planning, organizing, directing and controlling.

Typically, a nurse manager typically fulfils the role of a department manager or a unit manager. The nurse manager is responsible for long term planning in terms of cost control, nurse regulations control, nurse recruitment and handling of daily operations of a ward(s) 24 hours a day 7 days a week. In this manner, nurse scheduling and rescheduling are developed to form a competent team to accomplish the planning goal(s) of the nurse management system, which is to provide exceptional care services to patients with higher productivity.

2.1.2 Scheduling in Nurse Management System

Scheduling is defined as the allocation of resources in a defined pattern for the responsive performance of specific activities (Baker, 1976; Pinedo, 2002). Wren (1996) claimed that nurse scheduling in reality is personnel scheduling (Heizer & Render, 2006) with a decision making process (Morton & Pentico, 1993) that scheduling seeks to optimize in meeting the personnel allocation objectives. Quite often resources rotate throughout a schedule. As personnel scheduling deals with managing and monitoring the staffing needs over a specific time period (Heizer & Render, 2006) to balance customer demand, personnel work needs to be profitable (Ramli, 2004; Thompson, 1998). Thus, nurse scheduling can also be described as allocating a certain number of nurses to each shift over a certain time period subject to a set of constraints. These constraints are usually the working practice regulations and working preferences of nurses (Bai, Burke, Kendal, Li, & McCollum, 2007).

Burke, De Causmaecker, Berghe, and Landechem (2004) and Ernst, Jiang, Krishnamoorthy, and Sier (2004) asserted that personnel scheduling management should take several individual factors into account such as staff demand and availability, staff capability, staff desire to work, and labour costs to maintain appropriate customer-service levels. This is in line with Yano (2005), who suggested that scheduling also covers issues such as holiday and over time requests and various other rules related to the number of working hours, shift patterns, and nurse-to-patient ratio.

A nurse schedule provides a blue print of the working days and shifts, off days and holidays over a period of time. However, it is normally prepared on a weekly basis to avoid any uncertainty. As noted by Burke et al. (2004), it is a short term timetabling of staff with a typical horizon of not more than a few weeks. A more specific time horizon is suggested by Brusco and Showalter (1993), who suggested that scheduling could be for a period ranging from 1 to 8 weeks. In a nutshell, nurse scheduling is a nurse allocation process within particular periods of work.

The nurse scheduling procedure (Bellanti, Carello, Croce, & Tadei, 2004; Burke et al., 2007; Goodman, Dowsland, & Thompson, 2009; Gutjahr & Rauner, 2007; He & Qu, 2012; Ramli, 2004) is relevant to other sectors that deal with personnel service allocation such as transportation (Goel, Archetti, & Savelsbergh, 2012; Maenhout & Vanhoucke, 2010; Peters, Matta, & Boe, 2007; Weide, Ryan, & Ehrgott, 2010), hospital (Ahmed & Alkhamis, 2008; Augusto, Xie, & Perdomo, 2010; Wang et al., 2007), bank (Chandra & Conner, 2006), university (Abdul-Rahman, Burke, Bargiela, McCollum, Özcan, 2014; Abdul-Rahman, Sobri, Omar, Benjamin, & Ramli, 2014;

Burke, Mccollum, Meisels, Petrovic, & Qu, 2007; Burke, Petrovic, & Qu, 2006; De Causmaecker, Demeester, & Berghe, 2009; Kahar & Kendall, 2010), policing service (Engku Muhammad Nazri, 2001; Vila, Morrison, & Kenney, 2002), manufactory (Bhadury & Radovilsky, 2006), postal service (Qi & Bard, 2006), and hospitality (Burke & Soubeiga, 2003). The implementation of scheduling will also lead to rescheduling even in other industries. Therefore, it is important to understand rescheduling in the nurse management system.

2.1.3 Rescheduling in Nurse Management System

Rescheduling is synonymous with reactive scheduling (Bard & Purnomo, 2005a), real-time scheduling or re-allocation (Chiaramonte, 2008; Moz & Pato, 2007). The real-time nurse allocation is basically adjusting and implementing the nurse schedule to meet the demand of nurses between the units based on the number and acuity of patients (Brusco & Showalter, 1993). Research in the area of nurse rescheduling was started in 2003 by Moz and Pato. In commercial business sectors such as in manufacturing (Vieira, Herrmann, & Lin, 2000), airline (Ahmad-Beygi, Cohn, & Lapp, 2010; Bratu & Barnhart, 2006; Kakas, 2000; Rosenberger, Johnson, & Nemhauser, 2003; Thengvall, Bard, & Yu, 2000), railroad (Huisman, 2007; Potthoff, Huisman, & Desaulniers, 2010; Rezanova & Ryan, 2010) and healthcare (Clausen, Hansen, Larsen, & Larsen, 2001; Maenhout & Vanhoucke, 2011; 2013; Moz & Pato, 2007), the rescheduling application is more common. Nurse rescheduling is an important issue to be attended to as it has a direct impact on patient care and safety.

Unexpected events that cause schedule disruption include nurse(s) calling in sick or other urgent personal matters, assisting other understaffed units, handling impromptu outpatient care visitation, engagement of nurses in different health-related campaigns on a short-time notice, and other unanticipated events (Bard & Purnomo, 2005a; Chiaramonte, 2008; Clark & Walker, 2011; Moz & Pato, 2007). If these problems are not resolved in a collective manner, then such disruptions in the schedule may occur on a regular basis.

Notably, these disruptions may change the nurse management system which triggers rescheduling and hence affects the schedule performance (Dutta, 1990). Importantly, previous researches have argued that the change in the number of nurses due to disruptions often result in the deterioration of performance and therefore leads to the development of new schedules with little or major changes to the one formulated initially (Bard & Purnomo, 2005a; Chiaramonte, 2008; Maenhout & Vanhoucke, 2011; Moz & Pato, 2007). However, crisis gains opportunity, provided that quality change is implemented. According to Clark and Walker (2011), instead of small changes, radical changes to a postulated schedule maybe less problematic in the upcoming days due to the fact that short-lived actions and long-lived actions of Pinedo (2002) are judged differently and the nurse rescheduling may address numerous constraints so that the effective solutions for unexpected disruptions can be found.

Practically, short-lived actions are implemented to address minor disruptions placing minute effect on the service (Pinedo, 2002). For instance, the rescheduling involving short-lived actions may include recalling personnel from break, extending length of an individual's shift (overtime), or asking personnel to perform some additional tasks for a while (Engku Muhammad Nazri, 2001). On the other hand, long-lived actions are implemented due to major disruption(s) that result in substantial changes in the nurse schedule. These changes may consume more hours for processing and reassigning nurses with additional usage of resources (Pinedo, 2002). Such retrieve actions in rescheduling often result in additional work for some of the staff members, revocation of personnel leave(s) and reassigning of personnel to different jobs or shifts (Engku Muhammad Nazri, 2001; Thompson, 1999).

Bard and Purnomo (2004b) claimed that rescheduling some scheduled nurses must consider their previous shift's working condition. In emergency situations where nurse is required to work an impromptu shift, cost is not the foremost consideration but nurse coverage availability which abides in a ward regulation during rescheduling (Bard & Purnomo, 2004b, 2005a; Tang & Ghani, 2012). For instance, a nurse who is rescheduled to continually work right after his/her working shift (e.g., night shift followed by morning shift which no 8 hours difference), is more likely to be working with low vigour or absent for calling in sick. Thus, the head nurses may readjust the schedule again or head nurse has to work extra hours to cover the shortfall in such extreme situations. In this context, rescheduling results in little achievement in meeting the set objectives.

As the later part of rescheduling, nurse task assignment is considering a census matrix or patient classification system. Patient classification system is the most sophisticated technology for nurse rescheduling to determine patients' categories, which is based on the patients' health condition and severity of illness, and the estimated time required to care for a particular patient (Punnakitikashem et al., 2006).

Therefore, the patient should be given a new classification when his/her health or illness condition changes.

In a nutshell, rescheduling happens when an unexpected event occurs that mandates a scheduler to change the primary generated schedule with a minimum number of shift swaps to obtain an optimally workable nurse schedule. Thus, without rescheduling phase, it implicates a devoid of impromptu retrieval or replacement strategy to the nurse allocation in real-time. This interruption may give domino effect to the insufficient nurse coverage and unsatisfied nurse preferences while delivering service. As Dutta (1990) stated, the purposes of having rescheduling are: (a) to minimize any confusion and misappropriation from the working rota; and (b) to control and ensure that the needs of patients and care concerns are responsively addressed; and

2.2 Amalgamation of Nurse Scheduling and Rescheduling

The relationship between scheduling and rescheduling is supported by Vieira, Herrmann, & Lin (2003) and Pinedo (2002), who suggested that both functions are not independent of each other in real time. Although nurse scheduling is more of estimation and rescheduling is an implementation issue, this does not mean that they are separate issues. As mentioned earlier in Section 1.1, both require similar matters of consideration. This is further agreed by Pinedo (2002), who asserted that most of the time, scheduling process is actually a rescheduling process. Schedule may not run as planned because of the disruptions discussed above and other changes in the staff and patient forecasts. No schedule can be implemented entirely smoothly in the practical world without any changes made, which suggests that scheduling and rescheduling go hand in hand.

By combining nurse scheduling and rescheduling, nurse allocation can be made more comprehensively. With respect to handling and managing important ward covering problems, it is pivotal that the previous shift duties of nurses and their preferences are taken into account. In reality, mending or revising an impracticable schedule is tougher than building a new schedule in totality (Clark & Walker, 2011; Maenhout & Vanhoucke, 2011; Moz & Pato, 2007). Given that, restricting schedule changes may be affected by the postulated workforce coverage (Bard & Purnomo, 2005b; Chiaramonte, 2008; Moz & Pato, 2007) and cost (Bard & Purnomo, 2005a).

Secondly, schedule should also act as an additional buffer to cover any unexpected influences so that the total additions could be minimized and no other high-priority job gets delayed (Pinedo, 2002). Hence, overstaffing in schedules and cross-trained personnel are formerly safe side plans in scheduling (Thompson, 1999). However, understaffing condition is occurred in nursing working environment. Therefore, the preparation aspect of scheduling especially for real time implementation is still in a development phase. This dilemma of nurse coverage prompts this research to explore the linkage between nurse scheduling and rescheduling.

Furthermore, despite the difference of recovery time, short-lived actions would result in minimum changes or little impact on the original schedule compared to long-lived actions. Nevertheless, short-lived actions are usually handled manually by a nurse manager and are practically disregarded in nurse rescheduling problem, as others mostly go for impulsive retrieval (Bard & Purnomo, 2005b; Maenhout & Vanhoucke, 2011; McEachen & Keogh, 2007; Moz & Pato, 2007). Indeed, without comprehending the seriousness of the disruption to the postulated schedule before retrieval has clearly highlighted the weakness of rescheduling without leaning upon scheduling. However, this mutually dependency remains obscure.

Though the benefit of integrating scheduling and rescheduling has been emphasized in the theoretical arena (Clark & Walker, 2011; Vieira et al., 2003; Punnakitikashem, 2007) and empirical discussions (Chiaramonte, 2008), such integration has received little attention (Clark & Walker, 2011; Vieira et al., 2000). To date, there has been one most identical NSRP research so far out, that studied by Chiaramonte (2008). However, the pioneering study of Chiaramonte (2008) gave little attention on nurse preferences during rescheduling and addressed light disruptions using agent-based approach (AB).

As a whole, the combination of nurse scheduling and rescheduling is needed to overcome the less stringent work of scheduling which needs flexibility to match the nurses' capacity and preferences even in the period of uncertainty. Undoubtedly, a significant and responsive nurse scheduling with rescheduling system essentially facilitates and eases the supervisory work of a nurse manager.

2.3 Decisive Attentiveness of Head Nurse in Scheduling and Rescheduling

A head nurse holds a critical role in supporting and nurturing nurses to build a strong teamwork climate for effective patient care and assistance (Fletcher, 2001; Shirey, Ebright & McDaniel, 2008). But, he/she also is one the major factor that pushes nurses to leave (Taunton, Boyle, Woods, Hansen, & Bott, 1997). The head nurse has the authority to make changes based on the work priorities and areas requiring urgent support and assistance. As a result, such power and authority over nurses in connection to allocation and delegation have incurred serious complains. However, a nurse manager such as the head nurse has the responsibility in creating, formulating and amending a schedule because a quality nurse schedule potentially promotes a healthy working environment and hence good delivery of services (Aiken, Clarke, Sloane, Lake, & Cheney, 2009; Bellanti et al., 2004; Burke et al., 2004; Cohen & Golan, 2007; Eastaugh, 2007; Ford, 2012; Shahriari et al., 2014; Stimpfel et al., 2102). The following reviews the responsibility of a head nurse.

2.3.1 Attentiveness on Nurse Coverage

Nurse coverage relates to the number of nurses that are servicing a particular shift of a ward (Ramli, 2004). This includes the number of outsourced nurses from an agency or other third parties, although such practice does not promote a motivating working environment for quality patient care (Aiken, 2010; Dellasega, 2009; Ford, 2012). The head nurse should identify the appropriate levels of staffing in each shift to know whether the ward(s) has an ideal staffing, just-enough staffing or less-than-enough staffing condition (e.g., non-survival level).

Typically, nurse coverage and the total number of nurses required in each shift may vary. For instance, U.S. Bureau of Labor Statistics (2008) reported that a night shift duty has the least workforce demand compared to the morning shift and evening shift, as shown in Figure 2.1.

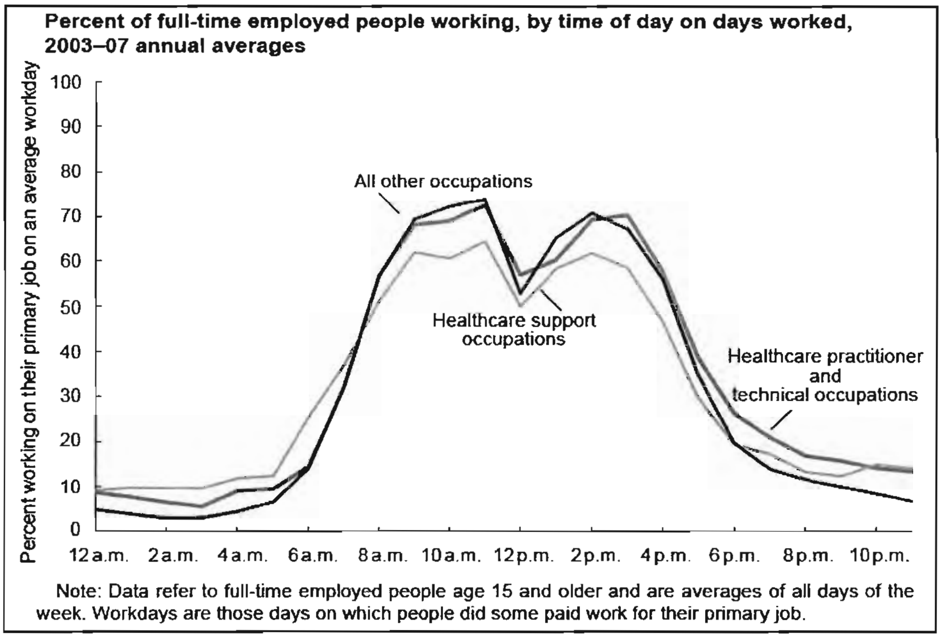


Figure 2.1. Percentage of average workload on a workday

Operations during the night play an important part towards providing a comprehensive 24-hour service to patients in every hospital. Indeed, hospitals go quiet at night as the patients mostly rest. In this situation, only a few staff members and nurses are required during the night shift (Horrocks & Pounder, 2006). Moreover, due to rising medical costs and shortage of nurses, hospitals are in short of staff during the weekends and late evening shifts (Landro, 2008). Therefore, using the night shift staffing as a basis, the head nurse may set guidelines or staffing benchmark to address the specific needs of nurses on a daily basis.

2.3.2 Attentiveness on Nurse Preference

The head nurse should also be well aware of each nurse's time/shift preferences to enable them to work responsively. By doing so, the nurses are being treated fairly. The head nurse should also ensure that the work life balance of the nurses is not affected which could potentially decrease nurse retention, as Cohen et al. (2009) argued.

2.3.2.1 High Preferences

A nurse preference in a schedule generally includes a consecutive shift pattern, equally divided shifts and workload amongst all nurses and little or no allocation of night shifts (Glass & Knight, 2010; Moz & Pato, 2007; Valouxis et al., 2012). But, nurse preference that regards to off days is seldom executed such as on-call compensation off days, constant off at the weekends, and request for off days. These off days are generally preferred by the nurses so that they can have quality rest time or be with their spouse and family members. Indeed, the off days are one of the significant factors impacting nurse depression (Chipas & McKenna, 2011; Murray, 2012) as well as spouse relationship ending up prematurely (McCoy & Aamodt, 2010; U.S. Centre for Disease Control and Prevention, 2009).

Nurse preferences are found to be of a less important consideration in understaffing especially when uncertainty is involved (Punnskitikashem, 2007). However, the head nurse should give support and consideration at all time (Gormley, 2011; Stephenson, 2014) as a sign of mutual respect towards staff nurses (AbuAlRub, 2004). For instance, the head nurse should plan the schedule by finding ways that complement ward regulations and nurse preferences. This is the reason why nurse preference is given a higher priority in this research. This is a vital consideration. As AbuAlRub (2004) argued, the importance of good interpersonal relationships between the staff and supervisors may potentially enhance security, mutual respect, and positive feelings which ultimately results in reducing stress. From the information technology

(IT) perspective, granting timely off days (e.g., nurse requested off duty, weekend off duty, days off during the public holidays, night shift compensations, and on-call compensations off day, etc.) depends on how smart the scheduling approach is used. It is ideal that a mutually acceptable schedule is developed. Kane-Urrabazo (2006) suggested that employees should be allowed to have their say in the creation of a schedule.

2.3.2.2 Fair Delegation

A head nurse is also expected to lead with fairness. The head nurse ought to formulate a just and balanced schedule, allocating justified shifts to every nurse so that potential disruptions, arguments, or disagreements from nurses in the future can be avoided (Azaiez & Al Sharif, 2005; Horio, 2005). However, fairness may be difficult to achieve when a disruption suddenly occurs (Clark & Walker, 2011). Though fairness is very subjective, allocating an equal chance of duty shifts and on-call duties as much as possible is important in (re)scheduling. This is because as Kane-Urrabazo (2006) remarked, trust and confidence in the head nurse and motivation to work in team can only be achieved through advocating fairness and consistency in the management and reward system.

2.3.3 Attentiveness to Nurse Uncertainty

A head nurse should also be aware of any changes in schedule due to uncertainty. Pamela (2007) stated that predictability and innovative scheduling model are essential components to increase nurse retention. Even though, practically speaking, it is difficult to run a postulated schedule, social network amongst nurses has proven to be the most important factor to support change and cover uncertainty in the work environment (Garrett & McDaniel, 2001). Even though the number of scheduled changes required to accommodate absenteeism changes is comparatively small, at times the nurses feel they are being unfairly treated because they to sacrifice their job preferences.

A supportive working climate may prevent nurses from being dissatisfied in their job. This requires partly that there is a good nurse schedule because the Queensland Nursing and Midwifery Office (2012) has underlined the risks of on-call disruption, as shown in Table 2.2. Hence, when making a schedule decision, it is vital that the head nurse consider the seriousness of the disruption (Maenhout & Vanhoucke, 2013), the impact of the radical change or non-radical change (Clark & Walker, 2011), and the quantity of changes in comparison to the original/postulated schedule (Moz & Pato, 2007). Constant on-calls give rise to patients' complains (Ministry of Health Malaysia, 2008), and the nurses can also suffer from exhaustion due to the tiring work schedule and unhealthy personal life. This also disturbs the social life of individuals which frequently result from disruptions (Cohen & Golan, 2007; Stimpfel et al., 2102).

Table 2.2

Risk Guide for Consecutive On-call Duties

Consecutive on-call shifts	Risk level	Controls
< 2	Acceptable	No immediate controls
2 or 3	Minor	Assess fatigue levels
4	High	Assess fatigue level and request support from other nurses and midwives to cover the on-call
> 4	Very High	No nurse will be scheduled for this number of shifts

In conclusion, a nurse manager plays an important role in influencing job satisfaction of nurses. If indeed the nurse manager's leadership styles affect the nurses' wellbeing, measures can be taken to develop mutually beneficial relationship that would lead towards efficiency, productivity and job satisfaction.

2.4 Appraisal of the Scheduling and Rescheduling Objectives

Over the years researchers have been trying to improve the problem solving parity related to nurse scheduling and rescheduling. In post 2000, total workforce size (Soubeiga, 2003), consecutive working days and off days (Bellanti et al., 2004), nurse labour cost (Wright, Bretthauer, & Cote, 2006) and unnecessary overtime cost (Azaiez & Al Sharif, 2005) are the key components that have been of a concern in the NSP. Moreover, due to absenteeism, NRP studies have considered the total workforce size to cover a ward operation, where schedule dissimilarity between the original schedule and the adjusted schedule is also evaluated (Moz & Pato, 2007; Punnakitikashem, 2007).

Given the above scenarios, cost-related objectives have become little less important areas of concern in scheduling and rescheduling. Instead, quality of a schedule is becoming increasingly more important (Clark & Walker, 2011). This is because pay-for-performance program is not a well-built strategy to increase quality of care in nursing, given that performance has been majorly defined in terms of staffing (Briesacher, Field, Baril, & Gurwitz, 2009). All of these reflect the displeasure of the nurses toward the work schedule (Chiaramonte, 2008; Wright et al., 2006). Hence nurse scheduling and rescheduling objectives are reviewed.

2.4.1 Objectives Based on Capacity

In order to let a ward operating at full capacity (i.e., power to produce nursing service), previous objective functions of personnel scheduling are evaluated by personnel cost of handling shortages (Bard & Purnomo, 2005c), unnecessary overtime cost (Azaiez & Al Sharif, 2005), and penalty on nurse coverage's constraint violations (Glass & Knight, 2010). In fact, capacity has not merely subject to nurse coverage, i.e. the number of nurses in a shift of a ward (Azaiea & Al Sharif, 2005; Bard & Purnomo, 2005a, 2005b, 2007; Bellanti et al., 2004; Brucker et al., 2009; Burke et al., 2007; Chiaramonte, 2008; Clark & Walker, 2011; Glass & Knight, 2010; Gutjahra & Rauner, 2007; Ramli, 2004; Valouxis st al., 2012), but also nurse competency which regards to the continuality of previous shift and current shift (Glass & Knight, 2010), and qualification and substitution of related skills (Azaiea & Al Sharif, 2005; Bard & Purnomo, 2005b). Pinedo (2002) and Lin et al., (2000) claimed scheduling and rescheduling processes are mutually related. To seek schedule quality, apparently, nurse coverage and nurse competency affect the effectiveness of capacity arrangement in the scheduling phase (Azaiea & Al Sharif, 2005; Bard & Purnomo, 2005a, 2005b, 2007; Bellanti et al., 2004; Brucker et al., 2009; Burke et al., 2007; Glass & Knight, 2010; Gutjahra & Rauner, 2007; Ramli, 2004) and rescheduling phase (Bard & Purnomo, 2005b; Punnakitikashem, 2007; Valouxis et al., 2012) respectively. Given that Pinedo (2002) and Vieira et al., (2003) claimed scheduling and rescheduling processes are mutually related, thus, nurse coverage and competency are the vital elements in capacity objective during the integration of NSRP.

In-depth study, covering signifies the difference between the minimum numbers of nurses required against the scheduled number of nurses (Warner, 1976). With regards to the objective of coverage, the least number of nurses needed in a ward was typically considered as hard constraint (Bellanti et al., 2004; Ramli, 2004). This is because nurse coverage objective that associated with constraints and regulations may denote the idea of utilizing the available nursing staff to avoid using extra staffing. Azaiea and Al Sharif (2005) suggested that nurse coverage could be adjusted according to nurse demand (e.g. number of nurse required differently in day shift and night shift). However, some studies have dealt nurse coverage with overtime factor, agency temporary staff, and pool nurse (Bard & Purnomo, 2005a, 2005b, 2007; Burke et al., 2007; Gutjahra & Rauner, 2007).

Bard and Purnomo (2005b) considered extra staffing to balance the contractual agreements and the managers' right to outsource outside nurses (primarily floaters and agency nurses). This was used to survive the operation of a ward during understaffing though it was costly. On the flipside, the numbers of staff for each shift could be maximized at all time in order to utilize the available staff, as overstaffing. In other words, they intended to constitute their regular basis of nurse coverage to an ideal coverage condition (Engku Muhammad Nazri, 2001; Ramli, 2004). However, the ideal coverage had been managed softly and even less achievable. Hence, targeting an intermediate stage of nurse coverage may be practical to real-world working condition. However, there is limited attention to intermediary coverage which resists survival coverage condition and further up to ideal coverage condition. As a whole, nurse coverage in scheduling and rescheduling is an important element for readying or coping real time crisis.

In the capacity that based on nurse competency, a nurse's nursing skill and vigour are counted-in when evaluating nurse performance. Nurse who works with renewed vigour depends on the rest given between his/her previous shift and current shift. Therefore, to seek quality schedule, the continuity in between the assigned shifts is constrained. For example, continuity that studied by Glass and Knight (2010) specified on connecting the last shift of a previous schedule with the first shift of the new current schedule. Next, another way of reflecting nurse competency in a schedule is taking nurse qualification into account, which in terms of nursing experiences and skills. Therefore, given a nursing shortage, supervision of senior nurse (Azaiea & Al Sharif, 2005) and skill-related substitution (Bard & Purnomo, 2005b) are constrained to achieve nurse competency.

2.4.2 Objectives Based on Preferences

Preferences in a schedule are generally based on shifts ordering preferences, personal desire or request on particular day, and fair treatment on shift assignment (Chiaramonte, 2008; Clark & Walker, 2011; Valouxis et al., 2012). In an objective function, these preferences is basically formed by calculating the total preference cost (Aickelin & Dowsland, 2004; Bard & Purnomo, 2007; Soubeiga, 2003) or soft constraint violations (Burke et al., 2006; Burke et al., 2007; Li, Lim, & Rodrigues, 2003; Ramli, 2004; Valouxis et al., 2012). Although preferences do not necessarily have to be considered in nurse scheduling, it plays a significant role in the nurse scheduling and even in rescheduling research.

Generally, several studies have included preferences into nurse scheduling problem (Aickelin and Dowsland, 2004; Bard and Purnomo, 2007; Bellanti et al., 2004; Li et al., 2003; Gutjahra and Rauner, 2007) and nurse rescheduling problem (Bard and Purnomo, 2005a; Clark & Walker, 2011; Moz & Pato, 2004). In 1976, Miller et al. were the first to formally address the nurse preferences scheduling problem. Chiaramonte (2008) also concluded that the preferences challenge has been gaining more research attention since 2000s. Especially in early 2000, Bard and Purnomo (2005a) presented a robust methodology to solve the preferences scheduling problem that can accommodate both the quantitative and qualitative details that nurse managers needed. Gutjahra and Rauner (2007) optimized the static assignment of nurses to demand on each single planning day by considering the nurses' preferences for hospitals and the hospitals' preferences for nurses. Additionally, the principal components of cyclic and preference scheduling were combined into one single model by adding preference constraints and demand constraints to work out the staffing shortage problem (Bard & Purnomo, 2007).

Nurse preference could be the offset point whenever nurse demand fluctuates. This problem gets worse in rescheduling that needs to endure frequent changes in the working environment. As nurse preference is often included in scheduling, it is not necessarily accounted in rescheduling (Chiaramonte, 2008; Punnakitikashem, 2007). Such working principle in rescheduling is to prevent any change that may deviate from the original schedule at a greater level (Moz & Pato, 2007), as the little change is more crucial than an ideal change. This is to reduce the outcome of unstable schedule, but undesirable schedule is intensified.

A non-preferable work schedule turns out to be a frustrating outcome (Valouxis et al., 2012). Generally speaking, nurses tend to prefer consecutive on/off workdays and dislike shift types such as night shifts. Therefore, such nurse preferences were then managed by constraint violation with some weighted penalty value (Choy & Cheong, 2012; Bellanti, 2004; Azaiez & Al Sharif, 2005, Bard & Purnomo, 2005a, 2007; Chiaramonte, 2008). In the terms of highly desirable preferences such as getting a timely off duty, previous studies have never explored the *tolerance* aspect between head nurse and staff nurses. Hence, very little evidence is available that effectively addresses a number of requests for 'off shifts' at the scheduling stage (Bellanti et al., 2004, Valouxis et al., 2012) and more so at the rescheduling stage (Chiaramonte, 2008; Punnakitikashem, 2007). This is because it is common practice not to involve staff nurse authority in the schedule making process, as highlighted by Bard and Purnomo (2005c).

Fairness, one of the preferred elements in scheduling, is defined as equal shift distribution among the available nurses particularly with regards to weekend shifts and night shifts (Choy & Cheong, 2012; Valouxis et al., 2012). Fairness in on-call nurse delegation has not practically considered in rescheduling problem. Perhaps, it is a difficult task to establish fair on-call delegation whilst satisfying higher preferences of nurses in producing fast retrieval solution (i.e. schedule). As affirmed by Clark and Walker (2011) that given small space to change in a postulated schedule during rescheduling is a hard task ever due to additional restrictions. In all, perseveringly reinforcing the preferences aspects (i.e., nurse preference and fairness) are needed for the sake of maximizing schedule's quality.

2.4.3 Objectives Based on Uncertainty

Most past studies have excluded real-time problems whilst developing their nurse scheduling approaches in the recent decades (Azaiea & Al Sharif, 2005; Bellanti et al., 2004; Gutjahra & Rauner, 2007; Li et al., 2003; Soubeiga, 2003). Perhaps, the complexity of scheduling problem is based on uncertainty (Burke et al., 2004). Once a schedule is implemented, adjustments are regularly made due to fluctuations in demand, absenteeism, equipment failures, and other unforeseen circumstances (Bard & Purnomo, 2005; Moz & Pato, 2007). Thus, uncertainty problem due to absenteeism can be denoted as imbalance between nurses required capacity versus the available numbers. As Punnakitikashem, Rosenberger, Behan, Baker and Goss (2006) argued that hospitals need to create a schedule that could help them manage critical understaffing situations by recruiting outsource nurses (e.g., part-time nurses or temporary agency nurses) and reassigning internal nurses (e.g., off duty nurses). In these manners, the objective functions of previous rescheduling researches were minimizing the expected excess workload of nurse (Punnakitikashem, 2007), the dissimilarity between old schedule and new schedule (Chiaramonte, 2008; Moz & Pato, 2007), and the cost of overtime wage (Bard & Purnomo, 2005b).

One imperative work of Moz and Pato (2007) in nurse rescheduling problem provides thorough details of various disruption instances that require handling the constraint violation effectiveness and time efficiency. Thus, we partially adopt the instances of uncertainty that may fit into our context of fortnight schedule for further recovery testing.

Importantly, a schedule might change frequently due to uncertain real-time problems. In this frequent change condition, a schedule becomes unpredictable and thus lost nurses' trust or not reliance on the postulated schedule. Therefore, scholars controlled the quantity of change to a postulated schedule during retrieval in rescheduling (Bard & Purnomo, 2005b; Moz & Pato, 2007). In fact, to achieve little changes of schedule adjustment, several aspects could be focused which are the dependency between scheduling and rescheduling, level of schedule disruption, and level of changes. Chiaramonte (2008) considered both scheduling and rescheduling problem, however, he showed less dependency upon both stages' decision where his rescheduling took the original schedule (output of scheduling) into account during making any retrieval decision, but not vice-versa. Perhaps, schedule's readiness for absenteeism problem could be optimized during scheduling as suggested by Clark and Walker (2011).

Next, the profundity of disruption level which is as the impact of a disruption to a postulated schedule could be considered. This is because by understanding how severe a schedule is disrupted, scheduler may not make impulsive retrieval decision during rescheduling (e.g., pre-retrieval or retrieval). Sometime, a postulated schedule might be able to endure a slight disruption. Lastly, level of changes in the retrieval process (i.e., radical change or non-radical change) has not been studied, though it is important in obtaining desirable schedule. Given that, retrieve accordingly such as abandon the spoiled postulated schedule and recreate a new one might be a pleasant plan (Clark & Walker, 2011). This aspect is used to pursue a quality retrieved schedule instead of only minimizing quantity change. Therefore, our research attempted to fulfil these three aspects of uncertainty objective.

In sum of the above appraisal objectives, besides the technique comparison (Brucker et al., 2010; Burke et al., 2007; Glass & Knight, 2010), capacity (Azaiea & Al Sharif, 2005; Bard & Purnomo, 2005b, 2007; Bellanti et al., 2004; Cai & Li, 2000) and preferences (Bard & Purnomo, 2007; Gutjahra & Rauner, 2007; He & Qu, 2012; Li et al., 2003) are two major research objective' focuses in static scheduling problems. The objective of personnel capacity and preferences were jointly targeted with partiality (Bard & Purnomo, 2005a; Clark & Walker, 2011; Gutjahra & Rauner, 2007; Valouxis st al., 2012) or separately targeted (Azaiea & Al Sharif, 2005; Chiaramonte, 2008; Li et al., 2003) in the nurse scheduling context. On the flipside, nurse rescheduling studies have been weaker in coalescing and adapting nurse capacity and preferences into uncertainty issue. Preference is a less profound objective in rescheduling and always assumed it has already accomplished in scheduling stage, thus can be ignored (Chiaramonte, 2008; Moz & Pato, 2007; Punnakitikashem, 2007). The preferences which subject to an approval of timely requested shift and compensated shift as well as its fairness are highly appreciated but rare to be accomplished. Moreover, lack of empirical efforts discussed the profundity of schedule disruption such as the dependency of scheduling and rescheduling, level of disruption impacts to a schedule, and level of schedule adjustment which subject to quality and quantity change. For that reasons, Figure 2.2 shows a vital facet of nurse scheduling and rescheduling problem (NSRP).

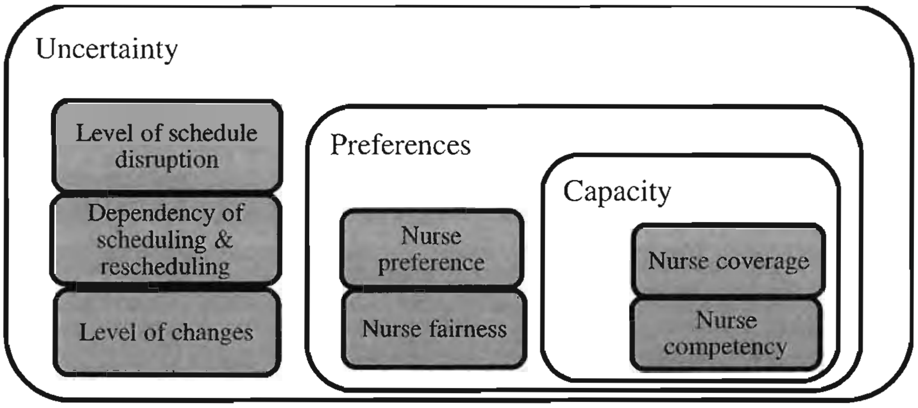


Figure 2.2

Facets of nurse scheduling and rescheduling problem

Figure 2.2 gives a glimpse on several criteria, we believe, that should be treated appropriately to formulate the objective function of nurse scheduling and rescheduling. The criteria are based on three main elements such as nurse capacity (Flynn et al., 2007; McEachen & Keogh, 2007), nurse preferences (Cheang, Li, Lim, & Rodrigues, 2003; Gordon, 2002), and nurse uncertainty such as absenteeism (Bard & Purnomo, 2005b; Chiaramonte, 2008). Based on the findings of the previous studies, we found that the core capacity is a subset of preferences and uncertainty. By fulfilling it, preferences are then followed. Uncertainty is a superset of preferences and capacity, whereby both are vital criteria for inclusion in the uncertainty.

2.5 Constraint Classifications

Generally, modelling constraints have been classified into two groups which are hard constraints and soft constraints. Hard constraint is designed to conform to strict policies; hence it always has to be satisfied. On the other hand, soft constraints are designed to conform to subjective personnel desires; hence it is not necessarily to be satisfied. To these extreme conditions, there is an awkward predicament which a semi-hard constraint can be relieved a bit from the over strictness (i.e., hard constraint), but has to be satisfied. In other words, the restriction has slightly loosened, though it is still important. This is because the extreme of restriction in hard constraints may be difficult to control the complexity of changes in real time working environment.

Therefore, additional classification with three different groups of constraints (i.e., hard, semi-hard, and soft) may seem to provide a better overview and control over the operation. Such line of thinking has been triggered by few researches (Muntz & Wang, 1990; Grandoni, Könemann, Panconesi, & Sozio, 2005) and also implemented by Kelemen, Franklin, and Liang (2005) and Abdallah and Jang (2014). Muntz and Wang (1990) suggested that timing controls was managed in both hard constraint and also soft constraint where cost is the aim of the study. Salido (2003) studied constraint satisfaction problem that separately gave ordering classification into each hard and soft constraint groups to improve efficiency check on the constraints. However, Muntz and Wang (1990) and Salido (2003) were unable to provide empirical evidence on implementing the semi-hard constraints and failed to provide a clear linkage between hard and soft constraints, though the need of semi-hard is emphasized.

In a study of constraint satisfaction, Kelemen et al. (2005) implemented semi-hard constraint as setting a flag to caution the deliberation element. Given that, semi-hard constraint had to be satisfied but eligibility could be earned before the sailor's new assignment whereas hard constraint must not violated but eligibility could not be earned before the sailor's new assignment. Moreover, soft constraint was countering sailors' preferences which could be violated, even if desired. Yet, there was an ambiguity on implementing the concept of semi-hard constraint, mainly no separate or definable code for semi-hard constraints but merely a plan of warns.

In nurse scheduling context, hard constraint basically reflects to obey organization's regulation and individual contractual agreements such as upholding nurse coverage (Clark & Walker, 2011; Gutjahra & Rauner, 2007) whereas soft constraint reflects some less important requirements that could not suspend a ward operation such as upholding nurse preferences (Moz & Pato, 2007). As a whole, hard constraints decisively influence a schedule's feasibility. However, due to lack of clear stance on semi-hard constrains, as far as research is concerned, no nurse scheduling or rescheduling problem has been solved with semi-hard constraint strategy.

2.6 Constraint Appraisal

Scheduling approaches have mostly included individual work constraints, staff preferences and hospital business rules in order to enhance roster acceptance and personnel job satisfaction at the workplace. Over the decades, the management of shift-work operations has become more complex. According to Coleman (1996), apart from the key issues of wages and benefits, workers generally have three main desires: better days off, better health, and more predictability in their schedule. So, there are various types of constraints while scheduling and rescheduling, as shown in Table 2.3.

Table 2.3

Summary of Constraints Types

No.	Constraints types	Sources
1	Basic schedule rule	Soubeiga, 2003; Li et al., 2003; Bard & Purnomo, 2005; Burke et al., 2003; Clark & Walker, 2011; Valouxis et al., 2012
2	Nurse workload	Burke et al., 2003; Azaiez & Al Sharif, 2005; Gutjahra & Rauner, 2007; Moz & Pato, 2007; Valouxis et al., 2012
3	Overtime	Li et al., 2003; Bellanti et al., 2004; Bard & Purnomo, 2005
4	Covering constraint	Bellanti et al., 2004; Azaiez & Al Sharif, 2005; Clark & Walker, 2011; Valouxis et al., 2012
5	Nurse skill classification	Soubeiga, 2003; Aickelin & White, 2004; Aickelin & Dowsland, 2004; Gutjahra & Rauner, 2007; Valouxis et al., 2012
6	Shift (types)	Li et al., 2003; Aickelin & White, 2004; Aickelin & Dowsland, 2004; Bellanti et al., 2004; Clark & Walker, 2011; Valouxis et al., 2012
7	Work consecutive/stretch	Bellanti et al., 2004; Burke et al., 2003; Azaiez & Al Sharif, 2005; Valouxis et al., 2012
8	Day off constraint	Li et al., 2003; Moz & Pato, 2007; Burke et al., 2003; Azaiez & Al Sharif, 2005; Clark & Walker, 2011; Valouxis et al., 2012
9	Shift ordering constraint	Bellanti et al., 2004; Azaiez & Al Sharif, 2005; Gutjahra & Rauner, 2007; Moz & Pato, 2007; Clark & Walker, 2011; Valouxis et al., 2012; Valouxis et al., 2012
10	Working weekend	Burke et al., 2003; Veen, Hans, Post, & Veltman, 2012; Valouxis et al., 2012
11	Daily adjustment rule	Bard & Purnomo, 2007; Clark & Walker, 2011; Chiaramonte, 2008; Moz & Pato, 2007; Maenhout & Vanhoucke, 2011

Constraints that are commonly set for creating a schedule are divided into two classes such as hard constraints and soft constraints. Nurse demand per day, per shift type, and per skill category are all usually hard constraints, while soft constraints are usually those involved with time requirements on personal schedules (Cheang et al., 2003). The goal is always for the scheduler to meet the hard constraints, while also aiming at meeting soft constraints. A brief explanation on each type of constraints is given below.

2.6.1 Basic Rule of Schedule

All full-time nurses must be assigned either 72 or 80 hours within a two-week planning period, depending on their contract. However, if a nurse is assigned fewer hours than specified in the contract, she must still be paid a full weekly salary (Bard & Purnomo, 2007). Therefore utilizing the available workforce to the fullest is important to reduce waste. For this, there are some basic rules. Hospital work requirements are basic rules to consider when making a schedule. For instance, no shifts must be left unassigned; each nurse must work exactly in only one shift-pattern per day. This means that a nurse should not work more than one shift. This is to ensure that the nurses (i.e. workforce resources) are able to fulfil the nurse demand reasonably.

2.6.2 Nurse Workload Constraint

Different labour contracts (full time nurses, part time nurses or floating nurses) may specify different work regulations, resulting in different agreed workloads. Nurse workload is either counted as the number of working hours or number of working days. In general, a full time nurse has to work six days in a week with one off day, or five workdays with two off days. In terms of work hour constraint, each nurse has to work on an 8-hour shift in 5 working days or 12-hour shift in 3 working days; no nurse may work more than 75 hours within a fortnight. Each nurse has to work at least 176 hours per schedule (4 weeks) which is equivalent to 14.67 days per schedule. The maximum labour time per week is around 36 hours. Due to the diverse work regulations in real problem, a nurse may have to work between 36 and 48 hours per week. However, this constraint does not include the special night shift's benefits and some other special days off in the particular week.

2.6.3 Overtime Constraint

This constraint is determined as an overload capacity, exceeding the amount of regular work times. For example, any additional working hours above 176 hours (regular working hours) in a fortnight schedule are considered to be overtime. In addition, a nurse is allowed up to four hours to exceed the number of hours for which they are available for their ward. This means that the maximum number of overtime periods that can be assigned to a nurse is denoted by the maximum number of overtime periods permitted. However, an overtime strategy may not be the best way to handle the understaffing problem. Scheduling models that consider a user's defined overtime constraints tend to focus on costs and not schedule quality or shifts arrangement. Moreover, Bard and Purnomo (2005a) strongly argued that more problems are created when nurses scheduled to work overtime, either voluntarily or mandatorily, call-in sick in a later shift.

2.6.4 Working Weekend Constraint

Weekend constraints have a greater impact on personnel satisfaction than other off days. When nurses take their weekend off, they can devote their time to their family. They may become upset when scheduled to work during the weekend. In some instances, the earliest weekend shift starts at 7 p.m. on Friday and the latest weekend shifts end at 4 p.m. on Monday, which becomes a problem for few. However, night shifts on Sunday are not considered as weekend shifts. As per general regulations, each nurse is to be given at least 4 days off during weekends in each 4-week schedule; at least 2 weekends off duty per 5-week period. This means that they should be given at least two weekend shifts in every two weeks. This constraint can be met in a rotation manner if and only if there is a large capacity of nurses. This is because of all nurses must have taking turn to claim the weekend off duty, conversely, number of available nurses who assigned on the weekends is decreased. This constraint violation may be exacerbated in rescheduling. Therefore, this constraint is rarely considered by the head nurse even though both parties know it is a highly preferred constraint.

2.6.5 Nurse Skill Classification Constraint

Nurses can be classified based on their qualification and experience. Apparently, nurses with higher skills and expertise are required to handle patients requiring critical care. For instance, nursing staff with a certain skill set is required in a ward. Or, nurses can be assigned to all shifts based on their qualifications and preferences. Or, a shift which requires certain proficiency can only be assigned to a nurse who has that skill. Therefore, the head nurse has more expectations from senior nurses, who are given more responsibilities. In this case, maximum capacity of available senior nurses is an important consideration for the head nurse.

Nurse substitution requirements ideally fall upon the nurse with higher skills and experience to work as a replacement. Therefore, experienced nurses have higher probability of temporarily replacing or aiding with personnel shortages in different units. Sometimes nurses with different skill categories may also be asked to substitute. This flexibility is applied in practice because of the work requirements in real time. Therefore, the head nurse should always plan to enhance the skills and abilities of her/his nursing staff.

2.6.6 Covering Constraint

Nurse coverage refers to the number of nurses needed in a shift of a ward which is also called nurse demand. Hence, nurse covering constraint is defined as the required number of nurses in each shift. The number can be exactly specified or it can be within a specific range. However, the number of nurses required daily may differ. Thus a robust scheduling approach is required to figure out the daily range of nurse demand in each work shift to allocate the available nurses across different wards.

The covering constraint is considered as hard constraint and reflects the research objectives. As shown in Table 2.3, in previous works the covering constraints have specified the decision variables that included the upper bound on the number of agency and on-call nurses. That is, a deviation of a maximum of one nurse is acceptable from the requested number of nurses in the night shifts (if four nurses are requested in a night shift, at least three nurses must be assigned for smooth operations). Other decision variables include limiting the total number of undesirable patterns to no more than a user-supplied parameter (P^{max}); coverage on each shift must range between lower bounds and upper bounds for each skill level; and discouraging the use of on-call nurses. However, some studies have highlighted flexible ways to distribute the available number of senior nurses evenly to each work shift especially to the night shift duty. Ramli (2004) fixed a number of nurses to handle night shift duty. Such approach may not necessarily work while planning in real time because the number of available nurses changes every day.

The issue of coverage in each work shift (by row in schedule) is as important as the daily coverage. These covering constraints intend to control each nurse's work pattern to make it parallel across a scheduling period, especially when allocating night shifts to each nurse. With regards to night shift constraints, the number of night shift per nurse can be represented either by a fix integer or a proportion of the nurse's total workload. For instance, the number of consecutive night shifts is at most 3, so each nurse must work at least 1 night shift per week. The maximum number of night shifts is 3 per period for 5 consecutive weeks; or night shifts must constitute at least 25% of total workload of each nurse. Each nurse must work one night shift per week and there must be at least n numbers of consecutive night shifts assigned to each nurse's schedule.

Moreover, each nurse is likely to be covering an equal number of work shifts (e.g., morning shifts (M), evening shifts (E), night shifts (N), and on-call duties etc.) to signify fair distribution. Indeed, fewer studies have focused on managing effective rotation of on-call duty. Expectedly, the difference in the number of nurses covering on-call duties may raise discomfort and frustration among the staff nurses and head nurse. In all, these constraints are mostly aggregated as shift arrangement issues.

2.6.7 Shift Type Constraint

Shift constraints depend upon how effectively or ineffectively a mix of various shifts is assigned to a nurse in a schedule. Practically, a shift can be categorized as multi-shifts set with variable start and finish time (Isken, 2004; Williams, 2008), strictly specifying as only one single shift (either day on or day off) (Narasimhan, 1996), day shift and night shift (Bard & Purnomo, 2005a), three different shifts (e.g., morning shift (M), evening shift (E), night shift (N)) (Li et al., 2003), or more than three shifts in a rotating manner. The variable start and finish time that has no generalized work time is also called irregular shifts. The shift times and durations are variable, which means there are no determined night-time or daytime shift. As a result, they are less likely to be used (Williams, 2008). Among these, it is also common to consider three different shift types in nurse scheduling (Burke et al., 2004; Wong, Xu, & Chin, 2014) in terms of work stretch constraints, work off constraints, and shift ordering constraints. All these restrictions and arrangements are important elements that need to be taken into account to generate an effective schedule.

2.6.7.1 Work Stretch Constraint

This type of constraint is restricting the number of consecutive working days with the range of between 2 consecutive workdays and 6 consecutive workdays. In other words, a single workday is not preferred. Unnecessary or extended work stretch becomes an issue in the nurse scheduling. As nurses become exhausted due to the extra work stretch violation, they may become less engaged in their duty. Hence, they are normally given free days or rest time after working consecutively. Besides restricting the overwork stretch assignment, pursuing an equilibrium distribution of different types of work shift (e.g., morning shift, evening shift, night shift) in a stretch is also observed by this constraint.

2.6.7.2 Off-day Constraint

Work off can be categorized into two groups: mandatory off days (public holidays, requested annual leaves, weekday off, weekend off) or permitted off days such as compensation off after a number of night shifts, on-call duty or consecutive workdays. The following are some work off violations stressed by previous studies on nurse scheduling. There are 2 off days for those with a 35-hour week, or only one off day for a 42-hour week. Officially, therefore, the mandatory number of off days given depends on the weekly workload. However, previous studies in table 2.3 had less perseverance to grant favoured off constraint. For instance, only 1 request for off day can be applied by a nurse in a fortnight schedule, with or without the assurance of approval. This is because the approval for the requested off days (e.g., off days on public holidays, annual leaves, and weekend off) depends on the sufficiency of the total number of nurses and their seniority in a ward. This issue has exacerbated nurse turnover.

Despite the optional wage benefit and mileage reimbursement, an off day is used as compensation when nurses are required for on-call duty, outpatient care duty and others while they are on leave. One extra off day (without pay) is given to reimburse one on-call duty when they are supposed to be off duty. Nevertheless, no empirical study has been conducted on on-call compensation. Besides that, nurses must be given at least a number of off days after a set of night shifts or after consecutive work time. For example, a 42-hour rest is required after a series of at least 2 consecutive night shifts, or 2 consecutive days off (at least 14 hours) to compensate 3 consecutive night shifts. However, there are some advance considerations for off duty preferences. For instance, no P.M./night shift before the requested off day and no A.M./day shift after the requested off day. The night off is significantly different from the requested off days because night shifts are often to be more tiring, leaving little energy and enthusiasm to spend and enjoy the following off day.

2.6.7.3 Shift Ordering Constraint

Shift sequence or shift ordering is combining different types of shifts in a sequential order. This ordering constraint is usually determined by mandatory break time rules, forward clockwise direction rule and some night shift concerns

The mandatory break time rule is to provide nurses with rest time from a heavy workload. This is the foundation of shift ordering constraint, such that 8-hour break shall be ordered between work shifts, or at least 16-hourrest between two consecutive shifts, or at least 11 hours of rest during any period of 24 consecutive hours. Typically, in 12-hours shifts basis, two work shifts are not allowed to be put in order. This is because it is against the labour law to impose 24hours of continuous working shifts. For example, no nurse can be assigned a shift from 7:00 a.m. to 7:00 p.m. and from 7:00 p.m. to 7:00 a.m. (day shift until night shift) or from 7:00 p.m. to 7:00 a.m. and from 7:00 a.m. to 7:00 p.m. (night shift to day shift).

With reference to 8-hour shifts, the sequence of assigning shifts is generally based on M<E<N<Off circadian rhythm (Ramli, 2004). This forward clockwise direction rule is to provide more rest time for nurses before they start the new day. For example, E→N work sequence is unfavourable than M→N work sequence. Furthermore, it is used to support a logical shift replacement for fine tuning purpose in rescheduling. For example, a nurse in an evening shift cannot aid a nurse in a morning shift of the same day ($E_d \leftarrow M_d$).

With regards to the mandatory night shift rules, nurses are not allowed to work N shift with a mandatory scheduled M shift the next day. Previous studies have stressed that such order needs to be considered as hard constraint (Clark & Walker, 2011; Ramli, 2004). Consideration of these set of constraints is to establish fairness in shift arrangement among nurses so that they receive sufficient rest time.

2.6.8 Daily Adjustment Constraint

Essentially, cyclical schedules are generally not effective in understaffing situations because of the regular changes (unexpected absences) of nurses. Use of external resources such as outside agencies and use of internal resources for on-call duties or for overtime are some of the common practices to address understaffed ward. Therefore, hospitals should develop a system to ensure nurses availability in order to avoid restriction rules of rescheduling when handling such uncertainties. For instance, nurses are not assigned to tasks on the days they are absent; head nurse is allowed to fine-tune at any day of a schedule; and isolated off day is unfavourable (Chiaramonte, 2008; Maenhout & Vanhoucke, 2011; Moz & Pato, 2007). These constraints are to be taken into account whilst repairing an ineffective schedule with considerable changes.

In resolving critical nurse coverage conditions, majority of researchers have focused on fairness perception of nurses being called, nurse's health condition, and the clockwise scheduling rule during shift replacement (Bard & Purnomo, 2007; Chiaramonte, 2008).

2.7 Nurse Scheduling and Rescheduling Techniques

Previous studies have shown significant results in addressing nurse scheduling problems by identifying 28 different categories of tools and techniques (e.g., Burke et al., 2004; Cheang et al., 2003; Ernst et al., 2004; Yi, 2005; Van den Bergh, Beliën, De Bruecker, Demeulemeester, & De Boeck, 2013). These techniques include optimization approaches (i.e. mathematical programming), constraint logic programming, constructive heuristics, expert systems, genetic algorithms, set covering/partitioning, simple local search, simulated annealing, tabu search, knowledge based systems, artificial neural networks and hybrid systems. In reviewing NSP from 1990 to 2000, Ramli (2004) clustered those techniques into four categories: optimization, search, constructive heuristics, and hybrid techniques. Since early 2000, knowledge-based approaches have gradually started to address rescheduling problem. This chapter reviews studies conducted after the year 2000, as shown in Table 2.4.

Table 2.4

Classification of Nurse Scheduling and Rescheduling Model by Solution Approaches

Authors	MO	H	MH	KB	HB	D	P
(1976) Miller et al.	MP						S
(1996) Berrada	MIP, IP		TS				S
(1998) Nonobe & Ibaraki			TS				S
(2000) Aickelin & Dowsland					GA+H		S
(2000) Cai & Li			GA				S
(2000) Dowsland & Thompson					TS+IP		S
(2000) Valouxis & Housos					IP+TS+LS		S
(2001) Brusco & Jacobs	ILP						S
(2001) Burke et al.			EA				S
(2002) Berghe					MA+TS		S
(2002) Burke et al.			VNS				S
(2002) Cowling et al.			HH				S
(2003) Burke et al.			HH				S
(2003) Dias et al.			TS,GA				S
(2003) Gutjahr & Rauner			ACO				S
(2003) Ikegami & Niwa					TS+B&B+H	/	S
(2003) Inoue & Furuhashi					EA+H		S
(2003) Li et al.					FCA+LS+TS		S
(2003) Soubeiga			HH				S
(2003) Yat & Hon			CP				S
(2004) Aickelin & Dowsland			IGA				S
(2004) Aickelin & Li	LP						S
(2004) Aickelin & White					GA+IP		S
(2004) Bellanti et al.			TS,LS				S
(2004) Isken	IP						S
(2004) Ramli					GA+LS		S
(2004) Topaloglu & Ozkarahan	GP						S
(2004) Winstanley					CLP+AB		S
(2004a) Bard	IP						S
(2004b) Bard	MIP						S
(2005) Akihiro et al.				NN			S
(2005) Azaiea & Al Sharif	GP						S
(2005) Brucker et al.		CH				/	S
(2005) Fung et al.					GCS/Simplex		S
(2005) Horio		CH					S
(2005) Matthews	LP						S
(2005) Özcan					GA+H		S
(2005c) Bard & Purnomo					CGB+IP+H		S
(2006) Bard & Purnomo	IP (B&P)						S
(2006) Belien	MIP (B&P)					/	S
(2006) Belien & Demeulemeester	IP(B&P)						S
(2006) Bhadury & Radovilsky		H					S
(2006) Özcan					GA+H		S
(2006) Suman & Kumar			SA				S
(2007) Aickelin et al.			EDA				S
(2007) Bai et al.					GA+SA HH		S
(2007) Bard & Purnomo	IP					/	S

Table 2.4 continued

(2007) Baumelt et al.			TS				S
(2007) Beddoe & Petrovic					CBR+TS		S
(2007) Bester et al.			TS				S
(2007) Brucker et al.		AC					S
(2007) Gutjahra & Rauner					ACO+DSS		S
(2007) Kelemci & Uyar					GA+LS		S
(2007) Thompson			LS,SA				S
(2008) Belie & Demeulemeester	MIP,B&P(CGB)					/	S
(2008) Burke et al.					VNS+HO		S
(2008a) Maenhout & Vanhoucke			GA				S
(2008b) Maenhout & Vanhoucke	B&P						S
(2008) Oughalime et al.					TS+GP		S
(2009) Goodman et al.					GRASP+CH		S
(2010) Brucker et al.		GL				/	S
(2010) Burke et al.			SS				S
(2010) Glass & Knight	LP						S
(2011) Kumara & Perera				GC			S
(2012) He & Qu					CP+CGB		S
(2012) Lu & Hao		ALS					S
(2012) Valouxis et al.					IP+LS	/	S
(2012) Veen et al.		H				/	S
(2013) Rocha et al.	MIP					/	S
(2014) Burke & Curtois	DP(B&P)						S
(2015) Wu et al.			PSO				S
(2002) Beddoe et al.				CBR			R
(2004) Moz & Pato	BLP						R
(2005a) Bard & Purnomo	IP(B&P)						R
(2005b) Bard & Purnomo	IP						R
(2006) Beddoe & Petrovic					CBR+GA		R
(2007) Moz & Pato					GA+CH		R
(2007) Punnakitikashem					IP+H	/	R
(2011) Clark & Walker	GP(CGB)						R
(2011) Maenhout & Vanhoucke			EA				R
(2013) Baumelt et al		GPU					R
(2013) Maenhout & Vanhoucke					EA+IP		R
(2008) Chiaramonte		AB					+

Abbreviation:

MO= mathematical optimization
H= heuristics
MH= meta-heuristics
KB= knowledge-based
HB= hybridization
D= decomposition strategy: /(applied)
P= problems: S(scheduling), R(rescheduling),
+(scheduling and rescheduling)

AB= Agent based
AC= Adaptive construction
ACO=Ant colony optimization
ALS= Adaptive local search
B&B= Branch-and-bound
B&P= Branch-and-price
BLP= Binary linear programming
CBR= Case-based reasoning
CGB= Column generation based
CH= Constructive heuristics
CLP= Constraint logic programming

FCA=forward checking algorithm
GA= Genetic algorithm
GC= Graph colouring
GCS= Guided Complete Search
GL= Adaptive heuristics-Greedy Local Search
GP= Goal programming
GPU= Graphics processing unit
GRASP= Greedy random adaptive search procedure
HH= Hyper-heuristics
HO= Heuristics ordering
IGA= Indirect genetic algorithm
ILP= Integer linear programing
IP= Integer programming
L= L-shaped method
LP= Linear programming
LS= Local search
MA= Memetic algorithm
MIP= Mixed-integer linear programming
MP= Mathematical programming

Table 2.4 continued

CP= Constraint programming	NN= Binary neural networks
DP= Dynamic programming	SA= Simulated annealing
DSS= Decision support system	SS= Scatter search
EA= Evolutionary algorithm	TS= Tabu search
EDA= Estimation of distribution algorithm	VNS= Variable neighbourhood search

2.7.1 Mathematical Optimization

Mathematical optimization is a selection of the best element from some set of available alternatives according to a number of particular needs or constraints (Azaiez & Al Sharif, 2005; Van den Bergh et al., 2013). Generally, there are various types of mathematical optimization techniques such as integer programming, linear programming, mixed-integer programming, and goal programaing which all are applied to maximize or minimize a single objective or multiple objectives' problems, respectively. Mathematical optimization techniques with set covering formulation in scheduling problems are able to achieve the lowest cost solutions (Azaiez & Al Sharif, 2005; Jaumard, Semet & Vovor, 1998; Matthews, 2005; Van den Bergh et al., 2013; Wright et al., 2006). On the other hand, nurse rescheduling problems are mostly handled by optimization techniques (Bard & Purnomo, 2005a, 2005b; Clark & Walker, 2011; Moz & Pato, 2004). Though they merely support minor changes, these techniques may face a heavy computational burden due to complex retrieval consideration with flexibility formation involved in rescheduling.

Previous single-objective mathematical programming (MP) studies had employed integer programming (IP) (Bard & Purnomo, 2005a, 2005b; 2007; Isken, 2004), linear programming (LP) (Brusco & Jacobs, 2001; Matthews, 2005; Moz & Pato, 2004), nonlinear programming for outline staffing ratio (Wright et al., 2006), and mixed-integer programming (MIP) (Bard, 2004b; Rocha et al., 2013) that only helped achieve one goal. Essentially, single-objective MP techniques have been used preferably in NSP since 1970s and 1980s (Cheang et al., 2003). Nevertheless, they involved more requirements when nurse tour scheduling problem was thoroughly studied.

Therefore, multi-objective mathematical programming (Azaiez & Al Sharif, 2005; Clark & Walker, 2011; Topaloglu & Ozkarahan, 2004) and decomposition strategies were used (Bard & Purnomo, 2007; Belien, 2006; Belian & Demeulemeester, 2008; Punnakitikashem, 2007; Rocha et al., 2013). Multi-objective MP is known as multiple criteria decision making that simultaneously optimizes more than one objective function. In comparing single-objective MP, the multi-objective MP considers more realistic aspects and has flexible objectives with priorities by weighting them (Cheang et al., 2003; Ernst et al., 2004).

Linear programming (Aickelin & Li, 2004; Matthews, 2005; Glass & Knight, 2010; Brusco & Jacobs, 2001) and integer programming (Isken, 2004; Bard, 2004a; Maenhout & Vanhoucke, 2008b) have been used extensively in NSP. When Aickelin and Li (2004) used linear programming that involved Bayesian optimization, they obtained a fairly close result as in optimal integer programming. Besides, the branch-and-bound method of linear integer programming is the well-known exact algorithm because the lower bounds can be found by linear programming relaxations or Lagrangian relaxations (Ernst et al., 2004). Ernst et al. showed that branching on constraints was more efficient than branching on single variables. However, the method has a lesser impact on exploration since it needs to be terminated once a few feasible solutions are found (Jaumard et al., 1998). Perhaps, this limitation has not hindered researchers in addressing a nurse rescheduling problem (NRP) which essentially can be thought as a matter of exploitation, or a repair function. Moz and Pato (2004) was the pioneer researchers in NRP who proposed binary linear programming founded on multi-commodity network flow formulations and tested it with real data. The integer programming of Bard and Purnomo (2005a) was tested on nurse skill considerations and more than one absenteeism day problem. Then, the branch-and-price method was employed by Bard and Purnomo (2005b) to upgrade the problem complexity by involving shift ordering preferences. Yet, computational time eventually turned out to be the usual drawback of these exact techniques. In other words, it is difficult to obtain solutions when larger instances are involved.

The branch-and-price (B&P) method has emerged as a powerful technique to aid a larger scale of integer programming (Bard & Purnomo, 2005a, 2006; Belien & Demeulemeester, 2008) and large scale of mixed integer programming (Belien, 2006; Belien & Demeulemeester, 2008). For instance, Belien and Demeulemeester (2008) used column generation technique to solve two different pricing problems repeatedly. The first one involved the generation of the individual roster lines which used dynamic programming. Mixed integer programming was then used for the second pricing problem to search for a surgery schedule, with a corresponding workload pattern that appropriately fit the generated set of schedule lines. Basically, mixed-integer programming (MIP) that mixed real-valued with integer-valued is used to tackle real complex problems (Bard, 2004b; Belian, 2006; Belian & Demeulemeester, 2008; Choy & Cheong, 2012). Despite the computational efficiency of the branch-and-price technique, it is less efficient in facilitating the integer variable when generating large numbers of integer variables. In Choy and Cheong's (2012) study, the MIP did perform well as it merely produced sub-optimal solutions. Their study focused on nurse staffing with consideration on skill and two fairness preferences of night shift duties and consecutive off days.

Azaiez and Al Sharif (2005) used 0-1 linear goal programming (GP) to resolve multiple objectives or priority issues in NSP. The objectives were regarding continuous service of nursing skills, staffing size, and fair considerations in the night shift duties, and isolated ON duty and weekend offs. However, they did not focus on the nurse preference element as it was categorized as a soft constraint. For nurse rescheduling, GP has lesser flexibility on schedule changes whilst tackling NSP (Azaiez & Al Sharif, 2005) and NRP (Clark & Walker, 2011). Clark and Walker (2011) showed the inflexibility of GP in approving late requested off days and other unexpected request for schedule change. Despite this, typical nurse preference was tackled by column generation in Clark and Walker (2011), even though the column generation was merely used to consider a relatively limited set of shift patterns.

In all, GP allows convenient sensitivity analysis because it is able to incorporate a few priorities with regards to nurse preferences; however, it places a significant burden on a decision maker to adjust selectively for a new non-dominated solution (Azaiez & Al Sharif, 2005; Clark & Walker, 2011; Topaloglu & Ozkarahan, 2004).

2.7.2 Heuristics

Heuristics is a problem solving approach based on discovering information and learning experience. It seems likely to search efficiently and produce approximate solutions in some optimization problems (Blum & Roli, 2003). Heuristics do not ensure the found solution is the optimality but guarantee best-so-far solution. Due to personnel scheduling is a NP-hard problem, heuristics were the most preferred approaches to addressing nurse scheduling problems in 1990s and they remain popular in the arena (Bhadury & Radovilsky, 2006; Burke et al., 2002; Lu & Hao, 2012). Berghe (2002) stated that in NSP basic heuristics are like shuffling neighbourhood, greedy shuffling neighbourhood, and core shuffling neighbourhood, which are good in healing the worst schedule. This is because they are all about exchanging some parts of a schedule with other parts of the schedule.

Besides, constructive heuristics approach is a step-by-step technique which constructs a solution based on a set of rules defined before hand. Hence, it has no initialization solution but consist of constructive phase and iterative phase that variables are managed in trial-and-error manner in order to satisfy every requirement (Burke et al., 2004). Horio (2005) developed a general project schedule based on the framework of Resource-Constrained Project Scheduling Problem (RCPSP/τ) that can be applied to various types of scheduling problems. Generally, RCPSP/τ had three main features. The first feature was the competency of the resources within a schedule horizon that varied at unit time. The next feature was the consumption of the activity within its duration time that varied at the unit level. The last feature was the generalized precedence constraints. This study managed to solve the 3-shift nurse scheduling problem (NSP) and satisfied all constraints.

Adaptive heuristics are boundedly rational strategies or a large class of simple rules of behaviour that lead to movement in apparently good directions (Hart, 2005). Adaptive heuristics have been applied to the nurse scheduling problem (Brucker et al., 2007; Brucker et al., 2010) and rescheduling problem (Chairamonte, 2008). They were a decentralize problem solving. An adaptive construction technique was used by Brucker, Burke, Curtois, Qu, and Berghe (2007) that enhanced the simple heuristics of Brucker, Qu, Burke, and Post (2005). Then, Brucker et al. (2010) decomposed NSP for the construction of shift sequences, schedule per nurse, and roster. Only high quality sequences and schedules were considered and selected accordingly. Later on, a greedy local search was carried out to improve the overall roster. The adaptive selections of Brucker et al. (2005) and Brucker et al. (2010) were applied to address Roster Boaster Instance problems and both managed to efficiently resolve not too complex requirements.

Chairamonte (2008) developed a real-time system that used an agent-based model for scheduling and retrieval purposes. However, a pool of nurses was needed in the study. This constructive heuristics merely tackled the modest schedule disruptions caused by more than three consecutive days' of absences and proved to be less effective in handling preference issues in terms of fair duty shift distribution, consecutive off days and requested off duty. One important outcome of this study was the need to amalgamate nurse scheduling and rescheduling. In a later study, Clark and Walker (2011) also were not able to fully address the problem.

2.7.3 Meta-heuristics

Meta-heuristics is a class of heuristics that combine basic heuristics in a higher level procedure to guide and search for reasonable solutions (Blum & Roli, 2003). Meta-heuristics are intended to deal with complex optimization problems that other optimization techniques have failed to be either effective or efficient (Van den Bergh et al., 2013). Even though hyper-heuristics are typically a higher level strategy than meta-heuristics, they are still heavily dependent on meta-heuristics as their input. Both have been used extensively to solve various nurse scheduling problems. Meta-heuristics techniques such as tabu search, simulated annealing, genetic algorithms, scatter search, constraint programming, and ant colony optimization are usually used to solve nurse scheduling problems and nurse rescheduling problems. In the late 1990's, tabu search (Nonobe & Ibaraki, 1998), genetic algorithm (Aickelin & Dowsland, 2000) and simulated annealing (Bailey, Garner, & Hobbs, 1997) have been proven to be sufficient in obtaining near optimal solutions for general nurse tour scheduling problems. Furthermore, the ORTEC benchmark instances of nurse scheduling problem were currently explored by modern meta-heuristics such as particle swarm optimization (Wu, Yeh, & Lee, 2015). In all, problem characteristic and complexity may be the factors to consider in determining which technique suits the best.

2.7.3.1 Tabu Search

Tabu search (TS) is a type of metaheuristic employing local search method. It moves iteration from one solution to another by searching and modifying some objective priorities in a neighbourhood space with the assistance of adaptive memory (Cheang et al., 2003; Berrada, Ferland, & Michelon, 1996). Based on previous studies of nurse shift scheduling and off days scheduling problems, tabu search-based technique was effectively exchanging day off cells with other work day cells for a particular nurse (Baumelt et al., 2007; Bellianti et al., 2004; Bester, Nieuwoudt, & Vuuren., 2007; Burke et al., 2006; Dias, Ferber, Souza, & Moura, 2003; Dowsland, 1998; Nonobe & Ibaraki, 1998). A list of prohibited moves was proclaimed tabu when the nurse was not available for the required category or when the shift was already assigned. Generally, the tabu list was updated throughout every iterations to avoid cycling the previous solution, which was in contrast with hill climbing.

According to Bester et al. (2007), although TS-based nurse scheduling is implemented in hospitals, the final schedule is still determined manually especially when some unpredictable absences arise. Thus, very little attention is paid to nurse rescheduling problem. Tabu search technique had found more accepted solutions in solving nurse scheduling problems by improving initial neighbourhood solutions (Bellanti et al., 2004; Dias et al., 2003; Oughalime, Ismail, & Liong, 2008), or tabu moves by adaptive memory strategy (Bester et al., 2007). For example, a slightly better performance of iterated local search in Bellanti et al. (2004) was noticed to prevent infeasible solution by means of greedy procedure in the neighbourhood. Their study was about handling parametric coverage requirements on consecutive working days and night shifts.

Studies have compared multi-objective mathematical programming with TS (Berrada et al., 1996) and meta-heuristics called genetic algorithm with TS (Dias et al., 2003), and local search with TS (Bellanti et al., 2004). In terms of computational time, mathematical programming was far much faster than TS in dealing with the nurse off days scheduling problem (Berrada et al., 1996) since TS is a heavy experimental implementation. Bellanti et al. (2004) took less than 8 hours for 60 coverage requirements. But, TS was more time efficient than genetic algorithm (Dias et al., 2003) even though it was slightly inferior in producing better fitness solutions (Bellanti et al., 2004; Dias et al., 2003).

2.7.3.2 Simulated Annealing

Simulated annealing (SA) is basically the generalization of Monte Carlo method for examining the equations of state and frozen states of N-body systems. According to Metropolis et al. (1953), the SA technique was inspired by the physical process of annealing that freezes liquids and metals into crystals. The slower the cooling schedule (rate of decrease) the nearer the direct algorithm to optimal solution will be. Kirkpatrick, Gelatt, & Vecchi (1983) claimed that SA was used to solve various optimization problems and was particularly effective in resolving circuit design problems. Bailey et al. (1997) used SA to resolve staff continuity.

SA is simple to formulate, requires less memory, and efficient. Thompson (2007) investigated the effectiveness of SA and local search-based meta-heuristics techniques. He found that they generally produced suitable nurse schedules, which could be evaluated by a weighted cost function that segregated the importance of each objective. Thompson tested the Sawing method and Noising method with SA. He demonstrated that Noising searched better schedules when the weight of constraints was set. In an in-depth study of SA, Suman and Kumar (2006) presented three SA-based techniques such as SA, SA with TS, and chaos simulated annealing (CSA) to search for a single objective optimization and multi-objective optimization.

In sum, SA is more time-consuming than TS. However, SA is less successful in finding a feasible solution when dealing with complicated problems, though this point-based methodis basically easy to use when handling mixed discrete and continuous problem.

2.7.3.3 Genetic Algorithms

John Holland invented genetic algorithms (GA) that was inspired by natural selection in the early 1970s. Processes observed in natural evolution were guided by the idea of survival of the fittest to overcome optimization problems. Typically, GA was applied to nurse scheduling problems (Aickelin & Dowsland, 2004; Burke, Cowling, De Caumaecker, & Berghe, 2001; Cai & Li, 2000; Dias et al., 2003) and further hybridized for sub-problems (Aickelin & Dowsland, 2000; Aickelin & White, 2004; Bai et al., 2007; Beddoe & Petrovic, 2006; Burke, Curtois, Post, Qu, & Velman, 2008; Inoue & Furuhashi, 2003; Kelemci & Uyar, 2007; Maenhout & Vanhoucke, 2011; Moz & Pato, 2007). GA hybridization such as evolutionary algorithms (EA) and memetic algorithm (MA) were embarked on slightly different ways of search within the construction of GA. Typically, EAs could be a potential enhancement technique of the traditional GA, given that GA alone has only used to tackle simple shifts scheduling problems (Aickelin & Dowsland, 2004; Cai & Li, 2000;), and slight complex tour nurse scheduling problems (Burke et al., 2008; Dias et al., 2003).

NSP of Aickelin and Dowsland (2004) was solved by indirect encoding based on permutations of nurses. In order to balance between solution quality and feasibility, it transformed the original problem into other rule-based problem then constructed the schedule step-by-step based on pre-defined rules. The indirect GA performed better than the tabu search. The results of the parameterized uniform order crossover (PUX) operator experiment suggested that more disruptive operators may practise better feasibility, but it may also affect the solution quality if it causes too much disruption. Hence, this appears as a question of balance between disrupting long sub-strings and inheriting absolute positions from parents.

Typically, a problematic search condition of GA was slightly infeasible but proved to be a highly suitable solution. In view of the fact that the role of penalty function was to avoid infeasible solution, it did not guarantee success when searching for appropriate feasible solutions. Mostly, the penalty weight setting was used merely as a rough guide that implicitly determines the quality of the final schedule at last. Burke et al. (2008) set the weights accordingly that enabled to define the importance of all required constraints. Likewise Dias et al. (2003) amended certain constraints' weight for further attention as repair unsatisfied constraint. Therefore, weight setting was a vital influence over a satisfactory or infeasible schedule. In all, repair intentions in fitness evaluation and operators were to transform arbitrary infeasible solutions into feasible solutions. However, as pointed out by Aickelin and Dowsland (2004), a simple and prompt repairing approach was difficult to find.

Next, three simple operators were developed to improve GA (Aickelin & Dowsland, 2004; Burke et al., 2008; Cai & Li, 2000). One advantage of GA is that it is not heavily dependent on any specific problem but focuses on the adaptive learning procedure via a number of phases, such as when Burke et al. (2008) employed two crossovers and three mutations to handle varying requirements. For the sake of careful exploration, Aickelin and Dowsland (2004) evaluated three different decoders (cover decoder, contribution decoder, and combined decoder) with four well-known crossover operators (e.g., PMX, uniform order based crossover, C1 crossover, and order-based crossover). Cai and Li (2000) tried to maintain diversity and avoid infeasible solution by using Hamming distance's crossover mask. As a matter of exploitation, repairing violated constraint at the end part of GA (Burke et al., 2008; Cai & Li, 2000) or even filtering initial search space to a feasible region were commonly worked in optimizing GA's solutions (Dias et al., 2003; Ramli, 2004). All this outlined the awareness for balance between exploration and exploitation in each operator. While more exploration added feasibility, it affected the solution optimality. In all, hybridization GA was then taken into account.

Based on the above discussion, constraint handling strategies could be undertaken by GA alone that are resolved by its representation (encoding), penalty functions, and operators. According to Aickelin and his colleagues, there is no pre-defined way of embedding constraints into GA (Aickelin & Dowsland, 2000, 2004; Aickelin & White, 2004). While feasibility could not be guaranteed, the results also showed that the randomness nature of GA led difficulties in constrained optimization in terms of constraint consistency. Thus, these shed light to some GA enhancements by hybridization, as discussed in Section 2.7.6.

While reviewing computational time, GA was not fast but it still provided feasible solutions within a reasonable time (e.g., about 10 minutes (Cai & Li, 2000)) since it is a suspicious search technique in generating (by operators) and filtering solutions (by fitness evaluation). In comparison between the search techniques, GA proved to be quite reliable in constructing work schedules with more flexible implementation (Cai & Li, 2000) but less time efficient compared to TS (Aickelin & Dowsland, 2004; Dias et al., 2003). However, GA and SA both turned out responsive in terms of computational time response. Though both were unable to produce optimal solutions in every run, they did provide good enough solutions (Bailey et al., 1997).

2.7.3.4 Scatter Search

Scatter search (SS) is a bit similar to memetic algorithms (MA) except the random decisions are replaced with intelligently designed rules and solutions are created by more than one parent (Burke, Curtois, Qu, & Berghe, 2010). To our knowledge, only one SS study was conducted to solve NSP. Burke et al. (2010) used scatter search to test previously published nurse scheduling benchmark instances and compared them with techniques such as constructive technique (Brucker et al., 2007) and memetic algorithm (Burke, Cowling, Caumaecker, & Berghe, 2001).

To compete with constructive heuristics of Brucker et al. (2007) and MA of Burke et al. (2001), the scatter search used variable depth search as an improvement method for the former and hill climber improvement method for the latter. The SS produced better solutions within less computational time. This may prove to be a robust and effective technique to solve related practical workplace instances.

2.7.3.5 Constraint Programming

Constraint programming is a programming paradigm where relations between variables that are stated in the form of constraints. In the 1990s, Cheng, Lee, and Wu (1997) and Weil, Heus, Francois, and Poujade (1995) noticed a number of studies using constraint programming (CP) to model the complicated rules associated with nurse schedules that involved cyclic and non-cyclic scheduling. This was because constraint programming (CP) had the propensity to work on highly constrained problems. Yat and Hon (2003) stated that CP can be a backtrack search technique, problem reduction with a vast numbers of constraints, and ordering heuristics that include value and variable ordering as developed in the graph theory. Nevertheless, meta-level reasoning is required in constraint generation to improve the search for a simplified NSP set (Yat & Hon, 2003).

Technically, constraint logic programming (CLP) is a systematically looped procedure that includes logic programming concepts for constraint satisfaction problem. Winstanley (2004) stated that high-level nature of logic programming was augmented by the seamless integration of one or more constraint solvers, thus making it easy to model the problem as well as constraints effectively. Although it is easy to express complex constraints and construct sufficient feasible solutions using constraint logic programming, it is not practically optimal (Cheang et al., 2003; Ernst, et al., 2004). Lack of effectiveness in finding an optimal or near optimal solution from a vast number of feasible solutions is a major drawback. As a result, Ernst et al. (2004) suggested to hybrid the flexibility of constraint logic programming with optimization techniques to conquer CP's weakness.

2.7.3.6 Hyper-heuristics

Hyper-heuristics is a high-level heuristics approach that adaptively chooses low-level heuristics (e.g., meta-heuristics techniques) to solve a problem. This generic method can also be applied to address problems of other domains. Only a few studies have employed hyper-heuristics in a nurse scheduling problem (Burke, Kendall, & Soubeiga, 2003; Cowling, Kendall, & Soubeiga, 2002; Soubeiga, 2003). Perhaps, the readied low-level heuristics was not necessarily the best algorithms but still performed well as per Soubeiga (2003) that pursued well-enough, soon-enough, and cheap enough solution across a wide-range of problems and domains.

Tabu search and genetic algorithm have always been in competition despite both being hyper-heuristics approach (Burke et al., 2003; Cowling et al., 2002; Soubeiga, 2003). Soubeiga (2003) applied choice function hyper-heuristics that compared tabu search and genetic algorithms for NSP with problem-specific information. Cowling et al. (2002) also conducted a comprehensive work, comparing hyper-heuristics with three meta-heuristics and integer programming. Both results proved that the choice function hyper-heuristics was more practical and offered more reliable solutions than direct and indirect genetic algorithms and tabu search. Unfortunately, Soubeiga (2003) affirmed that the hyper-heuristics only included a series of simple low-level heuristics.

Burke et al. (2003) proposed tabu-search hyper-heuristics for highly-constrained real world nurse scheduling problems. The high-level heuristics outperformed GA on feasibility rate but not to the highest optimality, since GA was exploited for problem-specific information. In all, hyper-heuristics can be a competitive technique with low-level heuristics due to its core vitality and usage to solve diverse scheduling problems.

2.7.4 Knowledge-based Approach

Knowledge-based approache relies on knowledge management activities to collect data on related experiences, and solutions are obtained by retrieving the data connected to the problem (Hsia, Lin, Wu, & Tsai, 2006). The whole idea behind this is to support and enhance the organizational processes of knowledge creation, storage, transfer, and application. It has been extensively used in the health care industry.

Neural network (NN) inspired from an intelligence of reproducing types of physical connections that occur in animal brains. Akihiro et al. (2005) used Hopfield neural network with binary neurons whereby the output took either the value of 0 or 1 as the new procedure for solutions which satisfied the indispensable requirements of a nurse scheduling problem. Even though combinatorial optimization problem such as traveling salesman problem can be solved by plane structured neural networks, NN was difficult to be applied to NSP because some NSP require a three-dimensional allocation. Therefore, they attempted to extend the plane structured neural network to the one that could take three dimensional structures in NSP and the result suggested that the constraints were limited to basic requirements.

To solve NSP, another similar kind of neural network technique is graph colouring (GC). Graph colouring of graph theory was applied by Kumara and Perera (2011) to typical nurse shift scheduling whilst considering the seniority factor. They divided shift groups via graph theory which was then expressed in an adjacency matrix. Different colours of vertices denoted different groups of nurses in the graph. Even though the allocation of resources was completed, the typical soft constraints that merely pursued better quality were completely excluded.

Generally, case-based reasoning (CBR) observed and stored scheduling matters in order to retrieve and perform repair moves whenever a similar situation occurs again. Hence, it was more likely to be used for rescheduling problem (Beddoe et al., 2002; Beddoe and Petrovic, 2006). Beddoe et al. (2002) tested case-based reasoning with complex real-world data collected from a hospital in United Kingdom. The core aim was to imitate how an expert human scheduler could produce a substantially smart schedule instead of following evaluation functions in other techniques. Beddoe and his colleagues further suggested hybridization of CBR by combining it with meta-heuristics techniques to resolve nurse scheduling (Beddoe & Petrovic, 2007) and rescheduling problems (Beddoe & Petrovic, 2006).

2.7.5 Decomposition Approach

Decomposition strategy is the design of algorithm wherein decompose a problem into various sub-problems. It is mostly needed for problems that are more complicated, or to tackle specific objectives, such as retrieval problem (Punnakitikashem, 2007), two stage problems with off days problem and shift types problem in cyclic tour scheduling (Rocha et al., 2013; Valouxis et al. 2012), Lagrangian relaxation of preference constrains followed by relaxation of demand constrains (Bard & Purnomo, 2007), and weekend shifts assignment (Veen, Hans, Post & Veltman, 2012). Generally, these decomposition strategies divided the problems into various sub-problems and then worked to resolve them through various techniques. In a nurse scheduling problem, the techniques incorporated were integer programming-based techniques (Bard & Purnomo, 2007; Belien, 2006; Ikegami & Niwa, 2003; Punnakitikashem, 2007; Rocha et al., 2013; Valouxis et al. 2012), and heuristics (Brucker et al., 2007; Brucker et al., 2010; Brucker et al., 2005; Veen et al., 2012).

Specifically, decomposition approach divides some sub-schedules in sub-periods of the scheduling horizon then solving them by assigning the highest priority activities to the highest priority employees. Furthermore, it also used to combine acceptable parts or remove drawbacks that associated with any individual approach (Ernst et al., 2004). Perhaps, the sub-problems division concept in decomposition is likening to a technique that consists of multi-level of operators, which each operator responds to a particular sub-problem. Thus, as we believe, evolutionary algorithm could also be explored for a flexible search by tackling different sub-problems.

2.7.6 Hybrid Techniques

More complex requirements ensue when amalgamating a nurse scheduling and a rescheduling problem in our research work and this has certainly pushed the study towards the superiority of hybridization technique. Since each technique has its pros and cons, the combination may perhaps produce better outcomes. As Burke et al. (2004) contended, there is a possibility of hybridizing early approaches (or some features of early approaches) with more sophisticated modern techniques to produce even better solutions. Hence, hybrid techniques are generally perceived to be more efficient to solve NSP.

2.7.6.1 Hybridization of Mathematical Optimization and Meta-heuristics

Essentially, mathematical programming (MP) is widely hybridized to solve numerous combinatorial problems (Cheang et al., 2003) particularly with regards to NSP (Aickelin & White, 2004; Bard & Purnomo, 2005c; Dowsland & Thompson, 2000; He & Qu, 2012; Ikegami & Niwa, 2003; Oughalime et al., 2008; Valouxis & Housos, 2000). Dowsland and Thompson (2000) and Valouxis and Housos (2000) attempted to solve a nurse scheduling problem by hybridizing classical integer programming (IP) with tabu search (TS). Dowsland and Thompson (2000) found that this hybridization considerably decreased the head nurse's burden from manually handling a time-consuming administrative task. In fact, when simple local search was used with tabu search, Valouxis and Housos (2000) further improved the solutions that were initially produced by integer linear programming.

Instead of producing provisional solution at the starting point, several studies that used mathematical programming and tabu search managed to exploit a global feasible region (Ikegami & Niwa, 2003; Nonobe & Ibaraki, 1998; Oughalime et al., 2008). Oughalime et al. (2008) hybridized tabu search and goal programming that used smart intensification to precisely aim for more fitting and faster output for shift ordering work and weekend preferences. Similarly, after a branch-and-bound integer programming had generated positive output for shifts scheduling problem, Ikegami and Niwa (2003) required heuristics to speedup the algorithm. However, sub-problems decomposition was needed to aid the IP which repeatedly satisfied constrains of nurse skill level, general nurse preferences and shifts balance distribution.

However, without tabu search, simple local search with different types of mathematical programming has also proved to work for nurse scheduling problems (Bard & Purnomo, 2005c; Valouxis et al., 2012) as well as rescheduling problem (Punnakitikashem, 2007). To deal with retrieval problem, Punnakitikashem (2007) integrated nurse rescheduling and assignment problems based on trade-offs between excess workload on nurses and staffing cost. He presented Benders' decomposition in stochastic integer programming for patient assignment followed by optimal greedy technique to solve recourse sub-problems. By using the same hybrid technique, Valouxis et al. (2012) won the INRC-2010 competition. Local search was incorporated into integer programming to tackle the first decomposed phase as off-days scheduling problem. Later, he employed integer programming to solve the last shift scheduling phase. However, Lu and Hao's (2012) result slightly outperformed Valouxis et al.'s when they attempted to solve the INRC-2010 instances by using adaptive local search techniques. Two unified neighbourhoods were implemented based on intensive search, intermediate search and diversification search.

Bard and Purnomo (2005c) added column generation based (CGB) to the hybrid IP and heuristics and formed a set of covering type problems by utilizing available nurses to quantify the benefits. In the study, the nurse preferences problem was modelled by integer programming and was resolved using the CGB technique that relied on intelligent heuristics for skill-related downgrading substitution option to identify better candidate solutions. As another hybrid CGB study, He and Qu (2012) took part in the ORTEC benchmark instances' competition with constraint programming based column generation (CP-CG). Firstly, it modelled problems by developing a column generation scheme following to which, CP paradigm took responsibility in pricing sub-problems as weighted constraint optimization problem. This hybridization was compared with four other hybrid techniques such as hybridized GA with LS, hybrid variable neighbourhood search (VNS), hybridized integer programming with VNS, and hybridized CP with VNS. CP-CG was found to be highly competitive with the hybrid IP. However, it had no diverse schedule produced and more computational time was required to derive optimal integer solutions at each tree node.

Fung, Leung, and Lee (2005) presented a hybrid constructive heuristics technique namely GCS/Simplex solver. This technique allowed Simplex method to be incorporated into the Guided Complete Search (GCS) framework to solve difficult nurse scheduling instances. This hybrid technique was a general constraint satisfaction problem resolver which addressed issues in terms of both computation time and number of failures. A special pattern occurred in the linear relaxed solution that allowed the Simplex method to determine a better value order for guiding the primary tree search solver in GCS towards a solution. Hence, GCS/Simplex solver was viable to solve different and cardinal constraints effectively.

Winstanley (2004) integrated constraint logic programming (CLP) and agent-based (AB) to solve nurse shift scheduling problems with partly self-scheduling. CLP-based hybridization included a high degree of user interaction and heuristics pre-processing. Nurses to be scheduled were denoted as semi-autonomous agents in the pre-processing. Hence, self-schedule was required before it could communicate with a global constraint solving CLP agent. In that case, schedule fairness might be affected by some aggressive agents since each took the responsibility to construct an individual initial assignment. However, the assumption of a collective and possibly negotiated agreement might not be valid in a real working environment.

2.7.6.2 Hybridization of Heuristics and Meta-heuristics

To achieve a balance between feasibility and optimality, Burke et al. (2008) and Goodman et al. (2009) worked on the construction and improvement of heuristics hybridization. Goodman et al. (2009) employed a look-ahead strategy based on knapsack GRASP model to ensure that the solutions constructed by the construction heuristics were easy to repair through local search. The variable in the knapsack model formed the basis of a feedback mechanism intended for diversification. Burke et al. (2008) employed heuristics ordering (HO) with variable neighbourhood search (VNS) for shift un-assignment and repair task. The HO method was applied to explore the search space. High quality schedules were found when they were combined with VNS. They also discovered that back-tracking was very useful in finding quick and better solutions by reducing poor quality solutions. However, it required a long computational effort that took more than one hour running time.

Indirect genetic algorithm was studied by Moz and Pato (2007) to overcome a nurse rescheduling problem. They tested real-life data instances by using hybrid constructive heuristics (CH) with several versions of genetic algorithms (GAs). The data instances were adopted and modified in order to test a two-week schedule. Technically, the CH was encapsulated by GA. The iterative reassignment of shift list (various way of task-to-nurse) was the responsibility of CH. GAs produced a new population that regarded two permutations of encodings such as list of tasks permutation and list of nurse permutation in order to perform a random shift list. Several GAs were experimented with different genetic operators for each encoding. Overall, the hybrid GA successfully solved the nurse rescheduling problem. However, in addition to the long computational time needed, another drawback was that it greatly failed to satisfy soft constraints.

Several studies mixed EA-based technique with other meta-heuristics in initial population of EA to solve a nurse scheduling problem (Aickelin & Dowsland, 2000; Aickelin & White, 2004; Bai et al., 2007; Ramli, 2004) and a nurse rescheduling problem (Maenhout & Vanhoucke, 2011; Moz & Vaz Pato, 2007). Also, some hybridization worked into mutation operator (Aickelin & Dowsland, 2000; Ramli, 2004) or after mutation operator (Bai et al., 2007; Kelemci & Uyar, 2007).

As the foundation of EA, GA is capable in searching a large search space. However, its limitation is less effective in identifying local optima which could affect computational time and quality of solutions. Therefore, several researchers have proposed GA hybridization to address this drawback. It was proposed that GA is hybridized with classical heuristics (local search, and variable neighbourhood search (VNS) etc.) (Aickelin & Dowsland, 2000; Inoue & Furuhashi, 2003; Kelemci & Uyar, 2007; Özcan, 2005; Özcan, 2006; Ramli, 2004), tabu search (Burke et al., 2001; Berghe, 2002), ant colony optimization (Aickelin, Burke, & Li, 2007), and simulated annealing (Bai et al., 2007).

Aickelin and Dowsland (2000) and Ramli (2004) both employed memetic algorithm (MA) (i.e. hybridization of GA and local search) where they tried to exploit sub-solutions to meet specific aspects of nurse scheduling problems. Ramli (2004) incorporated local search to meet nurse demand in the initial population and improve schedule quality through directed mutation. However, the nurse coverage requirements from each works shifts were low and timely nurse preferences were given little attention. Meanwhile, Aickelin and Dowsland (2000) applied nurse grade-based structure to define a hierarchy of sub-populations to build a partial solution into population. Then, a hill-climber in mutation was used to improve the solution. However, a heavy detection of problem specific knowledge was needed.

Few years after Aickelin and Dowsland (2000), Aickelin and Dowsland (2004) proposed an Indirect Genetic Algorithm with statistical comparison method. It primarily condensed results to a single value, even managed infeasible solutions through ranking method. In their study, Aickelin and Dowsland opted for indirect coding based on permutation of the nurses and a heuristics decoder that built schedules by these permutations. They found that an indirect GA proved to be more flexible and robust compared to Dowsland's Tabu Search in 1998. Nevertheless, the indirect learning technique of Aickelin and Dowsland was implicit and thus restricted the weight adjustment in the schedule building rules. Therefore, in 2007, Aickelin et al. embedded estimation distribution algorithm (EDA) in which an ant-miner technique was performed as a local search to emphasize high quality nurse-rule pairs. This was to identify building block directly and get better solutions. Because this technique was not fixed coded to certain instances, flexibility was enhanced.

Additionally, there were common ways of hybridizing within GA's operators. Kelemci and Uyar (2007) and Bai et al. (2007) added their intelligent paradigms after mutation procedure for further technique enhancement. Kelemci and Uyar study applied hill climbing to repair infeasible solutions in order to increase their success ratio and adjusted the weight of hard constraints which was difficult to be resolved, such as, days between two consecutive night shifts constraint. On the other hand, Bai et al. used multi-objective optimization techniques that needed stochastic ranking and selection system to deal with some constraints after mutation. Instead of the constraint handling capability, simulated annealing (SA) aided in its initial population to locate local optima.

The GA hybridization of Maenhout and Vanhoucke (2011) were highlighted on an initial population and crossover operator. In the initialization procedure, constructive heuristic was assisted by non-dominated Pareto optimality to diversify some high-quality provisional solution in a search space. Moreover, this evolutionary algorithm (EA) employed a crossover operator as a repair mechanism. It used network flow techniques and dynamic programming as improvement methods which revised and re-optimized a group of heterogeneous nurses. However, they disregarded the randomization principle in their crossover operator which basically only repaired the violation in order to maintain the original schedule in rescheduling problem.

The above review of EA hybridization shows that whether the hybrid techniques were dominated by other than EA (Beddoe & Petrovic, 2007) or encapsulated by EA (Berghe, 2002; Burke et al., 2001; Inoue & Furuhashi, 2003; Kelemci & Uyar, 2007; Moz & Pato, 2007; Özcan, 2005, 2006), we found EA to be a responsive operator. This conceptual optimization framework that consists of few operators can be implemented in a variety of ways with some degrees of sophistication. Conceivably, different effects ensue as a result of different combination types of technique as well as different order of a hybrid technique. According to Maenhout and Vanhoucke (2011), a skilled hybridization can produce higher flexibility due to different behaviours involved in it. In sum, the review pinpoints toward EA-based hybridization for tackling the complex amalgamation of nurse scheduling and rescheduling problem. The use of hybrid EA is considerably challenging in this regard because it was implemented to solve a nurse scheduling problem or a nurse rescheduling problem separately.

2.7.6.3 Hybridization of Knowledge Based and Meta-heuristics

In 2000s, hybrid case-based reasoning (CBR) was intended to work as a repair function for handling constraints violation in scheduling problems (Beddoe & Petrovic, 2007). However, they were not truly meant for rescheduling because the violations of rescheduling could be dynamic and not as static as in scheduling. Basically, CBR was used to repair those violated hard constraints through capturing and storing the scheduling knowledge and cases that had experienced by experts. The soft constraints of nurse preferences were not defined explicitly. Thus, to endure the implicit condition and increase the quality of solution produced by CBR, some meta-heuristics were integrated with tabu search (Beddoe & Petrovic, 2007) or genetic algorithm (Beddoe & Petrovic, 2006).

Beddoe and Petrovic (2007) implemented a tabu list of forbidden repairs in tabu search that reduced the search that was trapped in a 'loop' of repeating violations and repairs. But earlier, Beddoe and Petrovic (2006) developed GA to weigh and select the most important one from a large number of violating features when scheduling. Both studies showed significant improvement in the accuracy of CBR and reduced the number of features which needed to be stored for each problem. Nevertheless, the setting of chromosome weigh in GA must have well signified during fitness evaluation.

Li et al. (2003) used hybrid artificial intelligence (AI) for a class of over-constrained NSP. They used forward checking algorithm with variable ordering, non-binary constraint propagation, random value ordering and compulsory back jumping to handle hard constraints and then worked to improve nurse preference rule by tabu search and local search. This combination succeeded to produce solutions that satisfied all hard and preference rules to a greater extent within responsive computational time. This hybrid approach was based on a neighbourhood structure for vertical exchange of shifts.

Not much work had been done by using ant colony optimization (ACO) to solve nurse a scheduling problem. Theoretically, ACO was developed by Marco Dorigo in 1960. It was inspired by the ant behavior of finding food. Gutjahr and Rauner (2007) integrated ACO and decision support system (DSS) to solve high constrain problems. DSS was integrated to cover and balance unforeseen peaks of nurse demands by considering different preferences and costs. The result of the integration technique which consisted of simulation and optimization elements showed better improvement than the simple greedy technique. However, the shift scheduling problem was regarded as a static optimization problem and not dynamic. This implies that quick response to impromptu assignment was not considered.

2.8 Discussion and Summary

The above review suggests three main factors that have been investigated in the nurse scheduling and rescheduling problems. They are nurse capacity, preference, and uncertainty. Each has included many significant elements that can contribute to a higher quality of schedule using different techniques. Yet, there are still areas that require further study and attention. As Cohen et al. (2009) stressed, there is lack of flexibility in scheduling approaches. Following his recommendation, we focused on issues such as high nurse preferences, on call delegation issues, the dependency decision between scheduling and rescheduling that considers the seriousness of disruption, and the quality and quantity of change of retrieval. In order to cater for schedule flexibility with the addition of several constraints, meta-heuristics hybridization is a potential technique. As stated by Maenhout and Vanhoucke (2011), higher flexibility can be obtained by a skilled hybridization of different meta-heuristics behaviours.

MP approaches are effective in finding optimal solutions but there are still a number of difficulties with them. Besides the drawback of big numbers in integer variables' generation (Ernst et al., 2004), MP formulations is not friendly in expressing constraints and objectives. Hence, it is more commonly applied with simplified versions of the real-world scheduling problem or with few complications in the original problem. For instance, pricing problem usually becomes a real challenge if much of the problem complexities in the definition of the columns are ignored by a column generation method. In the case of involving flexibility and complexity that involves 2-dimension considerations in the scheduling and rescheduling problem, the advantage of using an exact method in the major problem may be lost.

Another limitation of mathematical optimization is its time consuming since personnel scheduling problem is a NP-hard problem. IP solver with integer constraints requires much more computational time. The amount of time to solve a family of related problems goes up exponentially as the size of the problem grows (Engku Muhammad Nazri, 2001; Pierce & Winfree, 2002). Thus, it can be assumed that mathematical optimization approaches may have the risk of failure in defining any feasible solutions even after a long computational time.

After reviewing a wide range of complex techniques in scheduling and rescheduling, EA-based hybridization seems to be outstanding. This is because EA is a stochastic search technique that can perform optimization by relying less on gradient information. This interest is driven mainly by the inadequacy of linear programming and other rule based systems while solving complex resource scheduling and rescheduling problems. Moreover, this population-based meta-heuristics tend to be relatively robust. It can produce reasonably good feasible (not optimal) solutions for any condition changes by merely incorporating certain problem specific information. At last, it is easy to be used in dealing complex objectives such as penalties for constraints violation. Therefore, EA could be a suitable pre-processing method of choice of hybridization to overcome the difficulty of identifying local optima.

CHAPTER THREE

EVOLUTIONARY ALGORITHM AND COOPERATION CONCEPT

This chapter considers the relevant theoretical basis of evolutionary algorithm (EA) and cuckoo search (CS). The taxonomy of EA and its principles are mentioned in Section 3.1 and Section 3.2, respectively. The structural components of EA are discussed in Section 3.3. Section 3.4 and Section 3.5 explain several types of hybridization structure and the rationale behind the hybridization. Since cuckoo search is applied to enhance evolutionary algorithm, Section 3.6 and Section 3.7 explain cuckoo search and its advantages. Lastly, a summary of the chapter is given.

3.1 Taxonomy of Evolutionary Algorithm

Evolutionary algorithm (EA) can be defined as a meta-heuristics which was inspired by the natural evolution processes (Maenhout & Vanhoucke, 2011). In the late 1990's, there was no precise identification of EA (Back, Hammel & Schwefel, 1997; Hertz & Kobler, 2000). The vagueness was probably due to two reasons: lack of powerful computer platforms at that time (Fogel, 1995) and some methodological shortcomings of the early approaches (Back et al., 1997). Back et al. (1997) noted that EA emerged in the late 1950s. After a few decades, this evolutionary computation had already started to attract attention in solving combinatorial optimization problems (Hertz & Kobler, 2000). But, hybridization of EA was encouraged to handle some difficult problems after EA failed to get an optimal solution (Grosan & Abrham, 2007). Later, hybridization was used to improve the performance of EA and quality of its solution (Sudholt, 2009; Zhang, Xu, & Gen, 2013). Earlier EA hybridizations are discussed in Section 3.5.

Evolutionary algorithms were originally developed from genetic algorithms, evolution strategies, evolutionary programming, and genetic programming (Back et al., 1997; Hertz & Kobler, 2000). Through various evolutionary computations, the four major approaches were independently developed by John Holland in 1975, Rechenberg and Schwefel in 1981, Lawrence J. Fogel in 1962, and John Koza in 1994, respectively (Grefenstette, 1986; Whitley, 2001; Yao, Liu, & Lin, 1999). Their problem solving from nature was so parallel that eventually in the early 1990 the different approaches were synthesized (Back et al., 1997), resulting in the introduction of EA, which was used in future studies (Ashlock, 2005; Burke et al., 2001; Grosan & Abraham, 2007; Maenhout & Vanhoucke, 2011; Zhang et al., 2013).

EAs implement self-adaptation and cooperation in each generation of population (Hertz & Kobler, 2000). The former indicates individuals evolving independently and the latter regards information exchange among individuals. Self-adaptation may dominate the classic evolutionary programming. According to Back and Schwefel (1993) and Yao et al. (1999), evolutionary programming with self-adaptive mutation typically is superior to evolutionary programming without self-adaptive mutation. Although they have slight differences in genetic algorithm, self-adaptation and cooperation are both essential. Indeed, a mutation operator of genetic algorithm can be implied as a self-adaptation process, while parent selection and crossover are the cooperation procedures. The operators of GA have a common conceptual base of producing individuals, to be discussed in Section 3.3. In this respect, GA remains the most renowned form of EA (Whitley, 2001). Hence, EA with GA-based is employed in this research.

Technically, evolutionary algorithm has emerged to combine classic heuristics in higher level frameworks since Blum and Roli (2003) acknowledged that this algorithm is a kind of meta-heuristic. Thus, several neighborhood local search algorithms could have involved, especially during the self-adaptation phase. Moreover, population-based method is another identity of EA (Hertz & Kobler, 2000). There are seven main features of EA in order to understand the philosophy of the population-based method. Hertz and Kobler (2000) summarized them as follows:

1. *Individuals.* They are problem specific where parts or sets of solutions can be initially formed as infeasible solutions or feasible solutions.
2. *Evolution process.* For the sake of survival of the fittest, a fixed size of population is evolved through steady state replacement (e.g., partly changes the population) or generational replacement (e.g., totally changes the population). In implementing evolutionary algorithm, parallel programming allows asynchronous evolution. Practically, create new individuals, select individuals to keep for next generation, and select individuals to thrown out from the current population are the actions in an iteration during the evolution process.
3. *Neighborhood.* Information exchange can be based on unstructured or structural population.
4. *Information sources.* The number of parents needed to create new individuals must be specified. New individuals can also be created on the basis of the history of the population.
5. *Infeasibility.* Infeasibility can be dealt with by several ways such as rejecting, penalizing, or repairing an infeasible individual.

6. *Intensification.* It is also named exploitation. In this self-adaptation phase, improved algorithm such as local search can be applied to each individual in the population.
7. *Diversification.* It is also named exploration. It is a noise procedure that randomly perturbs individuals to prevent premature convergence. The exploration has unexpected results that do not necessarily improve an individual.

Overall, evolutionary algorithm searches population-based problem solving solutions via learning process, randomizes information exchange process, and evaluates individual process (Whitley, 2001). The advantages of EA can be attributed to few factors such as population-based good enough solutions, machine learning, robustness, feasibility that gives quick approximate solutions, constraint handling and multi-objective optimization (Grosan & Abraham, 2007; Maenhout & Vanhoucke, 2011; Ramli, 2004). Additionally, flexibility and adaptability to the task at hand are both the most significant advantages of EA (Back et al., 1997). Therefore, with the aim of efficiently and effectively exploring a search space, EA can be considered as a generate-and-test search process pertains to exploration and exploitation.

3.2 Exploration and Exploitation

Fundamentally, exploration and exploitation are the two powerful elements in the search process of EA. Previous studies have referred to these elements differently. Exploration is also called diversification, while exploitation is known as intensification (Blum & Roli, 2003; Grosan & Abraham, 2007; Hertz & Kobler, 2000;

Sorensen & Sevaux, 2006). Exploitation is essentially searching neighborhood search space to get higher quality solutions. On the contrary, exploration moves to unexplored areas of the search space. Its purpose is to examine unvisited regions and generate solutions that differ in various ways from earlier ones. Hence, in our point of view, the main difference between exploration and exploitation is all about 'wide' search and 'deep' search.

Both are important because they may converge, leading to inaccurate solutions. Adaptive evolutionary algorithms have been built based on the principle of exploitation and exploration to avoid a premature convergence problem and optimize the final results (Blum & Roli, 2003; Grosan & Abraham, 2007). But, convergence speed can also be used as a form of feedback to alternate between the two modes. As pointed out by Al-Naqi, Erdogan and Arslan (2010), 'explore' mode occurs if the convergence speed is too slow. In contrast, 'exploit' mode occurs if the convergence speed is too high. Convergence is observed for the sake of tracking the domination of exploration or exploitation as well.

Therefore, a good search technique must find a good trade-off between exploration and exploitation in order to find a global optimum (Al-Naqi et al., 2010; Razali & Geraghty, 2011). A balance between exploring the search space and exploiting the best solution is a must. The balancing effort can be executed in various ways. For instance, in exploration, poor solutions must have a chance to go to the next generation, and in exploitation good solutions go to the next generation more frequently than poor solutions. This balancing effort is probably common in selection (Burke et al., 2001; Dias et al., 2003; Razali & Geraghty, 2011; Sharma & Mehta, 2013). Although evolutionary algorithm has operators to decrease or increase the diversity of population, most lack the means to control exploration of the population (Sorensen & Sevaux, 2006).

Additionally, the balancing effort involves the interplay of evolutionary operators with local search such as mutation and selection work in memetic algorithm (Inoue & Furuhashi, 2003; Sorensen & Servaux, 2006; Sudholt, 2009). However, the balancing effort did not work well when operating crossover. Slow convergence occurred when too much exploration that crossover creates randomness search points (Sudholt, 2009), or premature convergence occurred when too much exploitation that merely guides search point for repairing purpose (Maenhout & Vanhoucke, 2011). In all, these imply insufficient flexibility and adaptability to the crossing over tasks at hand. Back et al. (1997) noted that both flexibility and adaptability attributes are the most significant advantage of EA.

3.3 Construction of Evolutionary Algorithm

Generally, the fundamental concept of Evolutionary Algorithm (EA) is an imitating evolutionary process by combining solutions to produce better solutions thriving on the survival of the fittest. The construction of EA is illustrated in Figure 3.1 and Figure 3.2 shows the pseudo code of an EA.

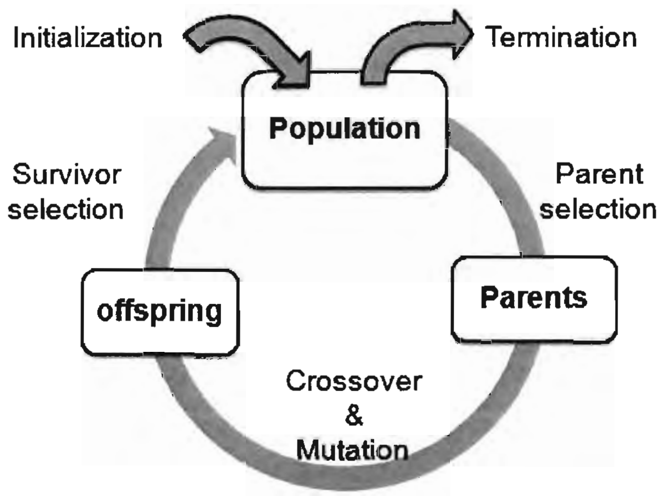


Figure 3.1. General construction of Evolutionary Algorithm

$t := 0$;
initialize $P(0) := \{\vec{a}_1(0), \ldots, \vec{a}_\mu(0)\} \in I^\mu$;
evaluate $P(0) : \{(\Phi(\vec{a}_1(0)), \ldots, (\Phi(\vec{a}_\mu(0))\}$;
while $(\iota(P(t)) \neq$ **true) do**
  recombine: $P'(t) := r\theta_r(P(t))$;
  mutate: $P''(t) := m\theta_m P'(t))$;
  evaluate $P''(t)$: $\{(\Phi(\vec{a}_1''(t)), \ldots, (\Phi(\vec{a}_\lambda''(t))\}$;
  select: $P(t+1) := s\theta,(P''(t) \cup Q)$;
  $t := t + 1$;
end

Figure 3.2. Pseudo code of Evolutionary Algorithm

Back and Schwefel (1993) outlined the evolutionary algorithm shown in Figure 3.2. In this formula, the evaluation process yielded a multi-set of fitness values. Moreover, $Q \in \{\emptyset, P(t)\}$ is a set of individuals that are additionally taken into account during the selection step for next generation. The following subsections explain in detail the incorporation of EA with problem-specific information.

3.3.1 Individual Representation

In EA, individual representations and the coding of the individual representations are devised in the form of a solution (i.e. Individual). Precise representation may provide better information. There are a few ways of structuring individuals such as by performing string representation (Grefenstette, 1986) and matrix representation (Ramli, 2004). The individual's coding called as data structure often indicated the gene of the evolutionary algorithm (Ashlock, 2005). Some of the general types of encoding are binary encoding, non-binary encoding, and permutation encoding. Among them, binary encoding is the simplest and initially used approach to represent the characteristics of a solution (Back et al., 1997; Grefenstette, 1986). It is simple because it only obtains two types of bits in a chromosome (e.g., 1101101, xxyyxxy, +--+-+-). However, due to its simplicity, it does not represent a real life problem. This is important because the processes of selection, crossover and mutation depend on the perceived performance of the individual structures as defined by the problem (Grosan & Abraham, 2007).

On the other hand, non-binary encoding uses real numbers or characters (e.g., 1234567, ABCDEF) to form chromosomes. Thus, a detailed representation of genes to a real problem is more likely (Deep, Singh, Kansal, & Mohan, 2009; Maenhout & Vanhoucke, 2011; Ramli, 2004; Whitley, 2001). This encoding may overcome the weakness of binary encoding because it represents reality to some degree. Permutation encoding is stressed once the gene's position of a string chromosome is taken into account such as in a sequential manner (e.g., 3412567). Therefore, besides string representation, permutation encoding has been stressed in edge representation for ordering matter such as a salesman traveling problem (Al-Dulaimi & Ali, 2008) or ordering task problem (Gupta & Dhingra, 2013).

3.3.2 Initial Population

At this stage, a population of candidate solutions is randomly initialized. Basically, this is the earliest stage of incorporating other heuristic techniques to produce better fitness solution (Khaji & Mohammadi, 2014; Whitley, 2001). In fact, randomization should be the focus in constructing evolutionary algorithms. Nevertheless, there exists non-random initial population study that obtained efficient evolution. For instance, Lin (2009) presented evolutionary algorithm with non-random initial population to plan the manipulator configuration along a path in its former stage. Once the optimal configuration is obtained by the evolutionary algorithm, the optimal chromosomes should be reserved as the initial population. Though the planned path is smoother than traditional GA, the drawbacks of the initial solution are that it depends much on unambiguous problem-specific information, which results some unexplored domains being lost. For these reasons, initial population with randomness attribute is much reliable to pursue diversification. To understand more about population initialization, the work of Kazimipour, Li, and Qin (2014) can be referred to.

3.3.3 Evaluation of Fitness

Evolution is the result of survival of the fittest (Ashlock, 2005; Grosan & Abraham, 2007). Hence, a fitness function rates the potential solutions by maximizing or minimizing their fitness. It judges the quality of evolved individuals and determines which individuals are fitter and better to bring to the next generation. This is an iteration procedure in which fitter individuals may soon approach to the optimal solutions. Thus, setting a penalty value (the input of fitness function) plays an important role here.

3.3.3.1 Foundation of Fitness Setting

Individual representation and fitness function are correlated. A fitness function must reflect a relevant measure towards a suitable representation to man an effective EA. Michalewicz and Fogel (2002) considered the interaction of the representation with a suitable selection strategy or search operators in light of the evaluation function. For example, the fitness of a given string is the number of positions at which it agrees with a reference string (Ashlock, 2005). Giving a penalty value to some violated hard and soft constraints is also a kind of an evaluation of fitness and it has widely been used in a nurse scheduling problem (Burke et al., 2004; Clark & Walker, 2011; Grosan & Abraham, 2007; Moz & Pato, 2007; Ramli, 2004). There is no one general standard set of penalty value, previous researches merely set the penalty value relatively based on the importance of the constraint. Perhaps the weights set are determined subjectively, thus leading to an incomparable condition as different nurse scheduling researches may have different sets of penalties. Besides, a schedule should be fair-enough (workload balance) to everyone and less disruptive to nurses' health, families or social lives (Azaiez & Al Sharif, 2005; Horio, 2005). However, being fair-enough is very subjective. According to Chen and Yeung (1992), as cited in Burke et al. (2004), the quality evaluation for shift work scheduling is based upon psychological adjustment, well-being, health, personnel and social problems, and performance and accidents.

In fact, setting up a penalty value for constraints violation based on its relative importance is an ever changing process. Kelemen et al. (2005) noted that situation changes may require frequent tuning of constraint satisfaction in order to compare effectiveness. Likewise, in some real time condition when a nurse demand of a ward is grown but nurses are limited, monitoring a nurse schedule becomes more restricted and complicated. This condition can be revealed by examining the schedule's fitness value.

3.3.3.2 Fitness Evaluation of Quality Schedule

Based on the above reasons, setting a more precise penalty value for evaluating a schedule quality is critically needed. Indeed, this constraint restriction strategy may signify fairness among nurses. Thus, as shown in Figure 3.3, the interaction between quality of life and quality of schedule is almost alike. Basically, the quality of life is engaged by Maslow's hierarchy of need theory.

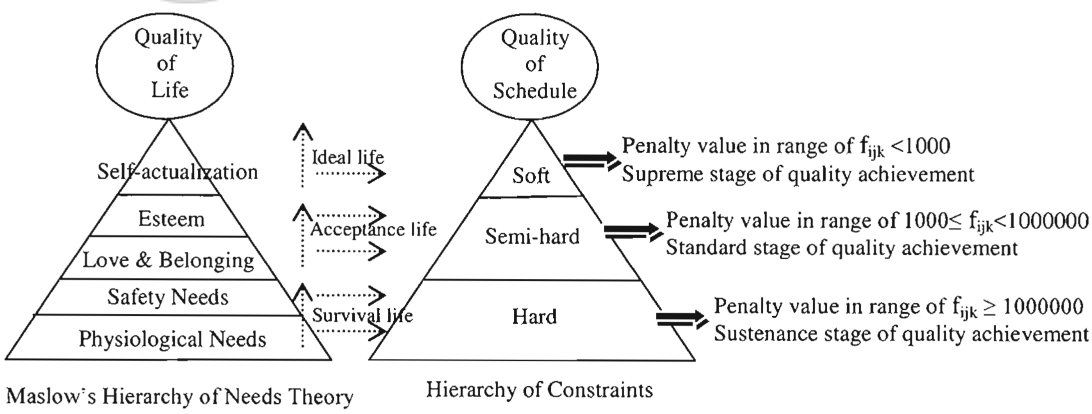


Figure 3.3. Interaction of Maslow's hierarchy of needs and hierarchy of constraints for quality schedule evaluation

Figure 3.3 illustrates how Maslow's theory inspires us in determining the three ranges of penalty value for each classification of constraint that results in a quality schedule. Essentially, Maslow's theory is a guide on how we develop and live our life better in reaching a state of 'optimal' mental health (Heylighen, 1992). Likewise, constraints aid to restraint a model in order to achieve an optimal solution. The five hierarchically organized levels of needs in Maslow's theory start with physiological needs, followed by safety needs, love and belongingness, esteem, and lastly self-actualization. The needs move to a higher level once the lower level needs are satisfied. As each level supports the next level, the most important class of constraints would fall at the bottom position as root, where the heaviest penalty values are set. Certainly, the peak position would get the lightest penalty value for violating soft constraints. This may explain 'why' and 'how' fitness calculation is handled in our minimization objective.

Furthermore, Mathes (1981) argued that the aim of each level of needs could be discussed and simplified into three main classifications whereby clarified the quality of life. The three classifications of quality living are started with a survival life, acceptance life, and lastly an ideal life (refer Figure 3.3). Basically, physiological and security needs in Maslow theory represent survival life. Likewise, in nurse scheduling and rescheduling context, the primary need is satisfying nurse regulations and operations of a ward even during an unpredicted understaffing condition. Secondly, people need family, friendships and work companions. Such social interactions induce a sense of belonging, mutual respect, self-esteem, acceptance and also strength for competence. The consideration of semi-hard constraints aims to achieve an acceptable schedule, which satisfies the preferences of majority of nurses.

It is deemed compatible to the ward's operation to achieve work expectation. A person who lacks a sense of belonging and esteem may be perceived as being helpless and weak. Correspondingly, a 'weak' schedule can be addressed by satisfying a nurse demand by paying attention to the need of the nurses that many people take for granted. When social needs are satisfied, staff nurses can encounter work challenges and stresses effectively.

Lastly, self-actualization is about growth need (Heylighen, 1992) where people wish to improve their overall personality by capitalizing on their potential. The need is specific and depends strongly on each people's desire to become an ideal role model. This need is achieved when the last class of soft constraints is not violated. In other words, ideally this is achieved when nurses can satisfy their preferences without going against the hospital regulations. Nurse retention might be achieved as a result.

3.3.4 Parent Selection

At this stage, EAs do not create new individuals. Parent selection operator is concerned with the way of selecting some kinds of individuals from the initial population that can potentially produce a good child for the next generation. The selection work resembles a search path in a search space. In other words, parent selection is deliberately made considering population diversity, selective pressure, problem space or population size, and convergence. Logically, higher selection intensity can be an advantage for large problem spaces (Legg & Hutter, 2005). However, small population size might lower the probability of enlarging population diversity. As a result, selection intensity may be compromised. In low diversity, ineffectively managing a small size of population and high selection pressure may lead to fast convergence (fast stuck in local optima).

Moreover, the question of how different the selected parents has been ignored in a mating strategy. In other words, does a pair of diverse parents have more potential to produce varying offspring than parents who look alike? This might be an important key of exploration in parent selection operator because it compromises flexible reproduction operators since most studies categorized a selection operator in an exploitation mode (Ashlock, 2005; Al-Naqi et al., 2010; Veerapen, Maturana, & Saubion, 2012). However, some studies overlooked its importance where merely copy a number of individuals to a mating pool without any selecting strategies (Tsai & Li, 2009; Yang & Wu, 2012). In fact, a parent selection operator is responsive to provide some potential permutation space for reproduction operators. Therefore, handling diversity of population is a significant task that has not only generally preserving by replacement strategy, but adapting the population diversity by parent selection. To control or manage a diversity search, this gives room to understand and enhance a parent selection operator.

Several well-known parent selection operators of evolutionary algorithms are discussed in the following subsections. They are tournament parent selection, roulette wheel parent selection, rank-based parent selection, and others. The basic concept is skewed towards elite parents, which act in exploit mode.

3.3.4.1 Tournament Parent Selection

Tournament parent selection is picking two most fit individuals each from different small groups that compete to become parents (Ashlock, 2005; Burke & Smith, 2000; Razali & Geraghty, 2011). The competition is intended to get better fit individuals who inherit from their parent for further exploitation. However, a selection too devoted to some subsets that appear to be promising, it may give negative effect (Hutter & Legg, 2006; Razali & Geraghty, 2011). As producing a variety of feasible offspring but which are less diverged from the potential parents. In fact, this type of selection has been used classically for solving some personnel scheduling problems such as nurse scheduling problem (Burke et al., 2001) and battalion rescheduling problem (Younas et al., 2013).

3.3.4.2 Roulette Wheel Parent Selection

Roulette wheel parent selection (RWS) is also known as proportional selection or fitness-based selection. This parent selection chooses parents in direct proportion to their fitness. In principle, a smaller fitness value deserves a smaller selected area and vice versa. RWS is quite a classic selection operator that proposed in year 1975, still, this type of selection operator has been applied favourably in recent nurse scheduling problem (Burke et al., 2008; Kim, Ko, Uhmn, & Kim, 2014; Moz & Pato, 2007). However, this selection method is not suitable if the characteristic of a population is ambiguous akin to big dissimilarity of negative fitness values (Mitchel, 1996; Deb, 2000).

3.3.4.3 Rank-based Parent Selection

Rank-based parent selection is similar to roulette wheel selection in that the individuals are differentiated before being selected arbitrarily as parents. In addition, parents are selected by ranking their fitness value (Cai & Li, 2000). Basically, ranking the individuals implies that an individual's chance of being selected gradually gets smaller or larger (Mithchel, 1996). The main purpose is to highly support the extreme elite parents to proceed with the mating process. Rank-based and tournament selections possibly outperformed proportional selection by maintaining steady pressure toward convergence (Razali & Geraghty, 2011). Hence, in order to solve nurse scheduling problem, this classical type of parent selection operator has been employed by Aickelin and Dowsland (2004) and Cai and Li (2000).

3.3.4.4 Others Parent Selection

There are other parent selection operators which merely studied theoretically, such as, fitness uniform selection strategy (FUSS) (Legg et al., 2004), Boltzmann selection (Mahfound, 2000), fitness uniform deletion (FUDS) (Hutter & Legg, 2006; Legg & Hutter, 2005), and adaptive selection (Veerapen et al, 2012). Basically, these approaches aim to preserve the diversity of population by clustering or managing the population individuals before they get selected. With the same intention, Maenhout and Vanhoucke (2011) applied pareto-optimal selection for solving nurse rerostering problem.

3.3.5 Crossover

Crossover is defined as combining or exchanging some useful groups of genes to produce new and fitter offspring than the preceding generations (Lewis & Paechter, 2004; Hertz & Kobler, 2000). There are various types of crossover and mating strategies formed in evolutionary algorithms that act as an exploration. This is because a crossover operator is highly trusted in EA performance.

Before studying some crossover strategies, crossover rate should be determined. The essence of crossover rate (*Pc*) is the intensity of introducing new solutions to the population. Srinivas and Patnail (1994) claimed that the higher the value of *Pc* the quicker the new solutions are introduced to the population. Therefore, *Pc* is defined as the probability *of* applying the crossover operator. Nevertheless, there are researches that attempted to maintain an acceptable level of population productivity throughout the process of evolution by setting *Pc* that approximately equalizes the probability of non-surviving individuals in a population (Pendharkar & Rodger, 2004; Lin, Lee & Hong, 2003). The non-surviving individuals are then replaced by offspring constructed by a crossover operator. The partial population (1-*Pc*) kept for the next generation is maintaining the diversity of population in order to prevent premature converges.

Basically, *Pc* is set based on the characteristics and complex interrelations with other aspects in the overall algorithm such as fitness value of both parent solutions (Srinivas & Patnail, 1994; Montgomery & Chen, 2010; Lin et al., 2003), population size (Montgomery & Chen, 2010; Lin et al., 2003), mutation rate and selection operator (Miki et al., 2000). Likewise, algorithm that is strongly elitist that frequently preserves superior solutions has more incentive to employ crossover to search more broadly. In this context, it sets the crossover probability at 1.0 (Montgomery & Chen, 2010; Eskandari & Geiger, 2008). Besides, Srinivas and Patnail (1994) suggested a solution whose fitness value is lesser than or equal to average fitness to be compulsorily undergo crossover (Pc=1.0). This is to escape from local optima. In other words, the extreme values of a crossover rate may be beneficial to the exploratory behaviour that focuses on the directionality of a search (Zaharie, 2009). In fact, this is a proper setting for some directed elements such as repair purpose.

To appraise a crossover strategy, different forms of data structures or representations and parents mating strategies are considered when deciding a well suit crossover (Maenhout & Vanhoucke, 2008a) such as string crossover operators, matrix crossover operators (Lewis & Paechter, 2004; Ramli, 2004), multi-parent crossover operators (Eiben et al., 1995; Porumbel et al., 2009; Ting & Buning, 2003) and parent-centric crossover operators (Ballester & Carter, 2005; Deep & Thakur, 2007; Garcia-Martinez, Lozano, Herrera, Molina, & Sanchez, 2008; Lozano, Herrera, Krasnogor, & Molina, 2004; Raghuwanshi & Kakde, 2006). Since our solution is a two-dimensional matrix representation, matrix crossover operators are the main center of attention here. In general, two orders of crossover which are either horizontal (cross-point cuts between numbers of row) or vertical (cross-point cuts between numbers of columns) are commonly used for a matrix chromosome. Besides the orders, the position of genes changed by some type of crosspoint is another vital consideration. For example, single point crossover, multiple point crossover, and uniform crossover are well-known conservative approaches to matrix representation.

3.3.5.1 Single Point Crossover

Essentially, a single point crossover is the simplest type of crossover. Since it is done by one cut point, only one side of the cross point (e.g., left or right, above or below) is swapped between parents, as shown in Figure 3.4. Apart from the order of crossing, this crossover operator has been specifically named as C1 (Reeves, 1996), 1PX (Holland, 1992; Kellegöz, Toklu, & Wilson., 2008), NBOP (Aickelin, 2000), and DBOP (Inoue & Furuhashi, 2003). There are cross points which are chosen at random (Bai et al., 2007; Kellegöz et al., 2008) and by purpose (Ramli, 2004; Aickelin & Dowsland, 2000). In row-wise crossover (Ramli, 2004) and grade-based crossover (Aickelin & Dowsland, 2000), only the representation structure either in a string form or a matrix form differentiates these two single point crossovers.

3.3.5.2 Multiple Point Crossover

A multiple-point crossover or *n*-point crossover is the natural extension of a single point crossover, where two or more cross points are chosen at random and the segments between them are exchanged (Eshelman, Caruana, & Schaffer, 1989). Based on previous studies, a multiple-point crossover is known as 2PX (Murate & Ishibuchi, 1994), and OBX (Syswerda, 1996). The researchers showed that multiple-point crossovers are more compatible in long length chromosomes. Apparently, they produce more variety of offspring that might be active in exploration as well. Figure 3.4 illustrates the formation of offspring done under two factors, i.e. the number of crosspoint (e.g., single point or multiple points) and the direction of crosspoint (e.g., horizontal or vertical) in a matrix crossover.

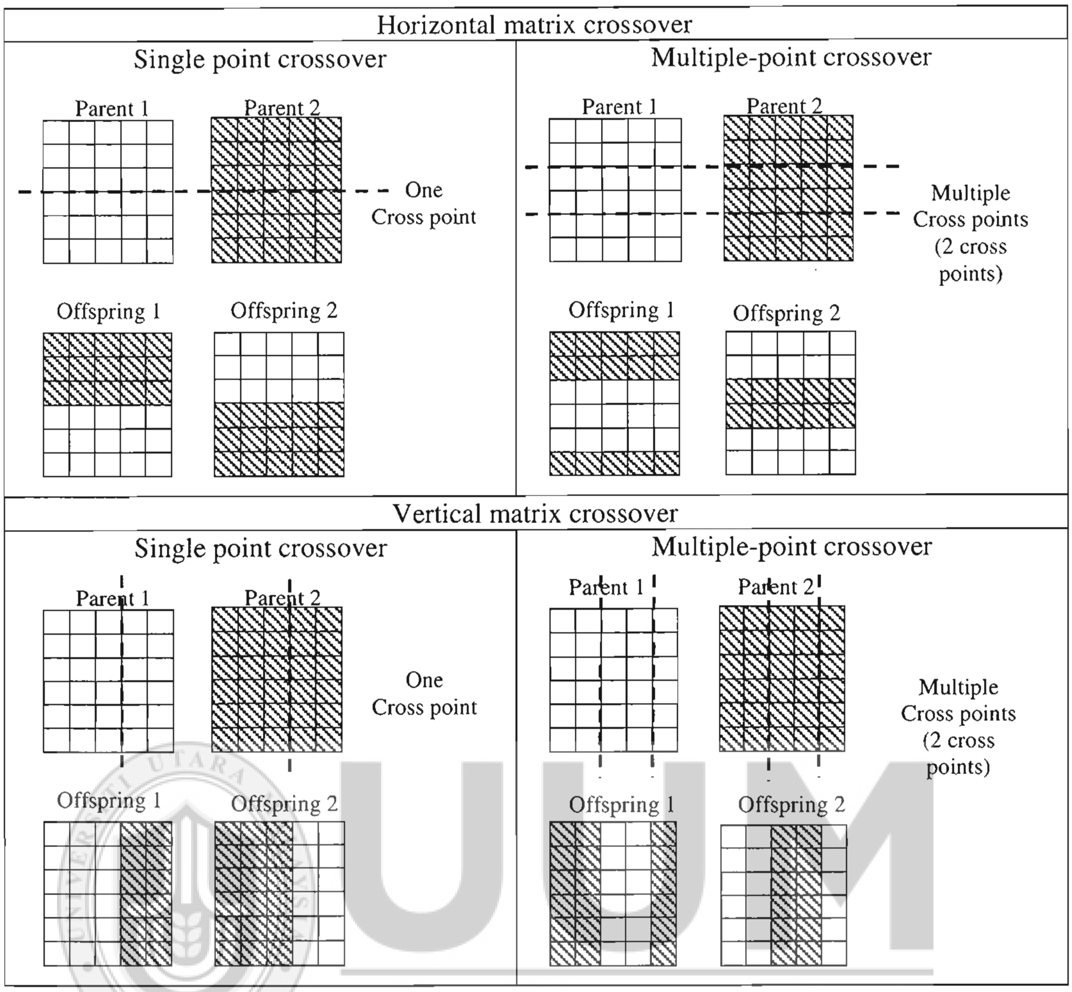


Figure 3.4. Types of single and multi-point matrix crossover

3.3.5.3 Uniform Matrix Crossover

Basically, uniform crossover that mixes at the gene level works more flexible and effective than a single or two-point crossover that mixes at the segment level. Cai and Li (2000) affirmed it to be a multipoint crossover. This is because of the random construction of a binary vector (e.g., 1001010). It indicates an equal chance (standard setting is 0.5 probability) of bits swapping as well as eliminating the problem of representational bias (Ashlock, 2005). Conceptually, the uniform crossover is a building block destroyer and the extra random numbers needed are computationally expensive. However, Aickelin and Dowsland (2004) believed in uniform crossover because it maintains a higher proportion of absolute positions of genes. But still, it shall not cross in large block since improvement may be gained by flexible disruption as well. In previous researches, uniform crossovers are named as PMX (Aickelin & Dowsland, 2004), UOX and PUX (Syswerda, 1996), hamming distance based on differ genes of strings (Cai & Li, 2000), and randomly selected nurse-based crossover (NBRS) (Aickelin, 2000; Burke et al, 2001; Dias et al., 2003). Figure 3.5 illustrates the general formation of each offspring during a uniform matrix crossover.

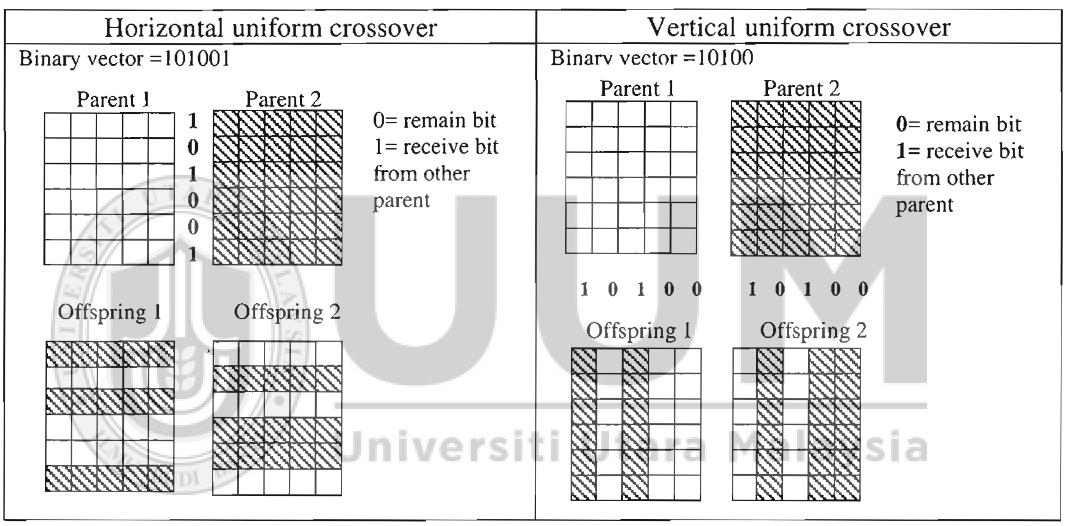


Figure 3.5. Types of Uniform Matrix crossover

3.3.5.4 Other Matrix Crossover

There are other unordinary matrix crossovers such as specific crossover operators (Maenhout & Vanhoucke, 2008a) and sector-based crossover (Lewis & Paechter, 2004). The specific crossover operators attempt to select cross points deliberately with respect to some case-specific constraint, or allow the construction of the feasible individual by exchanging or removing the worst or best genes between parents. Such crossovers examples are single parent specific crossover (Jan, Yamamoto, & Ohuchi, 2000), and nurse-based crossover with tournament selection (NBTS) or named as best placed events crossover (Burke et al., 2001; Dias et al., 2003). Though this fastidious exploiting problem-specific concept is able to remain promising sub-solution, it would also easily cause earlier convergence due to the loss of exploration in producing new child.

Lewis and Paechter (2004) proposed a sector-based crossover for a university course timetabling problem. Offsprings are produced by randomly selecting a sector of parent 2 and inserting them into parent 1, with the condition of placing the same gene position as in parent 2. This crossover technique transfers a larger number of genes per crossing over which signify larger steps in the search space than the other conservative crossover operators. However, the fixed position comportment may force an unfit situation that gives insignificant disruption to parent 1. Furthermore, the offspring might have unbalanced heritage from both parents, where the center of attention mostly fall upon one parent rather than the other parent. Although it allows randomly wrap around the sector to prevent bias genes' selection and tends to offer greater flexibility throughout future generations.

As a conclusion, exploration has been stimulated by crossover operator. A more disruptive operator may help enhance feasibility, but it also may affect solution quality if there is too much disruption (Aickelin & Dowsland, 2004). Without disregarding the diversity of a mating strategy, an attempt to inherit good genes from parents is focused at this stage (Maenhout & Vanhoucke, 2008a, Cai& Li, 2000). This might be a good way to lessen unnecessary disruption that causes infeasibility. However, less exploration may result premature convergence because randomization is still the core principle of a crossover operator. In that sense, a difficulty of a crossover in a two-dimensional representative chromosome is pressed on the less effectiveness of exploitation that an ideal search should able to flexibly move around feasible and infeasible domains. The difficulty was faced by Ramli (2004) in a nurse scheduling problem and Moz and Pato (2007) in a rescheduling problem. Since randomness of a crossover operator could violate hard constraints easily, Moz and Pato (2007) could only use a nurse-based cross point in a crossover operation. Meanwhile, Ramli (2004) applied a row-wise crossover based on the nurse skill level. Given that both fixed horizontal cross point behavior had limited the exploration, a more flexible way of crossing over with regard to various cross point strategies are needed.

3.3.6 Mutation

Crossover and mutation operators are called variation operators or reproduction operators or secondary search operators. Both have the same objective which is to produce new solutions. However, mutation is slightly different from crossover operators is that mutation operators are applied at the self-adaptation phase. Mutation intends to alter an individual itself with small changes by a swapping strategy, *k-opt* neighborhood strategy and others (Hertz & Kobler, 2000; Rice, 2004). It makes some noise or bulk by increasing the variety of genes to an individual in order to enlarge the searching scope for next generation (Li et al., 2008; Rice, 2004). Therefore, the small changes are able to increase the diversity which prevents individuals from a premature convergence towards local optima. Mutation operator is also known as a fine local tuning operator. Deep and Thakur (2007) stated that the proportion of population undergoing mutation and the strength of mutation are two paramount issues in mutation operator studies.

Grosan and Abraham (2007) claimed that the mutation rates (P_m) may be adapted to prevent premature convergence and to speed up optimization. Thus, the mutation rate is one of the concerns to drive the convergence toward the best solution. Reeves (1993) agreed that mutation is typically applied to lower probability (<0.01) than crossover probability since too small a value of crossover probability is associated with a loss of diversity and premature convergence. However, Grosan and Abraham (2007) recommended adjusting the mutation rates at P_m= 0.6 (if convergent) and P_m = 0.05 (if not convergent). Haupt and Haupt (2004), and Bremermann (1958), as cited in Michalewicz and Fogel (2002) claimed that the mutation rate may depend on a number of binary variables. In a nutshell, the mutation rate may not be held constantly because the mutation rates recommended based solely on experimental evidence.

Apart from the mutation rate, there are a few types of mutation in evolutionary algorithms. In encoding, there are binary encoded mutation (e.g., 0↔1) and permutation encoded mutation (e.g., a real number gene is swapped with another real number gene). Essentially, there are merely changing positions in between the same individual. However, the permutation encoded mutation which directly aims at altering positions changes less the child than the binary encoded mutation does (Wagner, 2004). Perhaps, binary EA is just changing a bit from a 0 to 1 and vice versa (Cai & Li, 2000; Haupt & Haupt, 2004). Therefore, the binary encoded mutation is more free or random in inverting selected bits.

In ways of perturbation, basically there are random (uniform) mutations and non-uniform mutations (light and heavy mutation) as directed mutations. A random mutation is when a gene is replaced with a random value between its lower and upper bounds. On the other hand, in a non-uniform mutation, the step size decreases as the generations increase. This makes a random search in the initial space and very little at the later stage (Haupt & Haupt, 2004; Ramli, 2004; Michalewicz & Fogel, 2002). The existence of directed mutation is probably due to a drawback that a new solution which generated by the conventional mutation has closed to its parent but may be far from better solutions. Since a uniform mutation does not utilize any global information extracted from a current population (Zhang et al., 2005), therefore, directed mutations probably have their own way of mutating guidance rather than relying solely on randomization. Readers may refer to Berry and Vamplew (2004), Danciu (2003), Ramli, (2004), Temby, Vamplew, and Berry (2005), and Zhang, Sun, and et al. (2005) for their proposed directed mutation.

3.3.7 Replacement Strategy

The objective of replacement is to maintain the same population size for each generation. Studies have focused on a number of individuals being replaced in each generation. In the early research, generational reproduction was normally used. There were enough amount pairs of selected parents to replace the entire population (Ashlock, 2005). However, the drawback of generational reproduction is that it may ignore or miss out some potential individuals after one extreme replacement. For that reason, steady-state reproduction is more likely to be used since it counts each single act of selecting parents and placing the children in the population as the next generation.

With regards as who shall be replaced, there are a few replacement techniques that are commonly used. They are absolute fitness replacement and elitism. Absolute fitness replacement is replacing the least fit individuals of the population with better fit children (Michalewicz & Fogel, 2002). On the other hand, elitism is exhibited when the individuals of a population with the highest fitness are guaranteed to survive in an evolutionary algorithm. Those guaranteed individuals are called elite (Ashlock, 2005). In fact, elitism guarantees a population with a fixed fitness function that cannot slip back to a smaller maximum fitness in later generations. This affects the current elite to be more likely to have more of his children in the future, causing their genes to dominate the population. For not undergoing any of the extreme elimination, there is a median or mod elimination in which eliminating individuals who are too close to an existing individual of the population (Lacomme, Prins, & Ramdane-Cherif, 2005; Porumbel et al., 2009; Ramli, 2004). This may preserve unbiased diversity.

3.3.8 Stopping Criterion

This stage of evolutionary algorithm involves a decision to either proceed with or terminate a searching process. Basically, the termination conditions stop after a fixed number of trails (Lacomme et al., 2005), when the search has plateau (Aytug & Koehler, 2000; Ramli, 2004), or when a fitness threshold has achieved which an expected solution is found (Ashlock, 2005). The fixed number of trails is set accordingly by a model user. Typically the parameter has to undergo a bunch of experimental testing. On the other hand, a plateau condition is a state of slight or no change following in a number of generations, such as three consecutive generations (Ramli, 2004) and 10 consecutive generations (Aytug & Koehler, 2000). Next, the generating process can be terminated when an expected solution is found. Mostly, this type of stopping criteria is used to search or fix an individual's fitness value according to a fitness threshold (Ashlock, 2005), making it appropriate for a repairing function. In general, the last two types of stopping criteria (i.e., plateau and expected fitness threshold) are basically suitable for searching within a boundary due to their termination depends on the achievement of an incumbent solution.

3.4 Hybridization Architectures of Evolutionary Algorithm

Basically, there are three essential types of hybridization architectures which are concurrent architecture, transformational hybrid architecture, and cooperative architecture (Grosan & Abraham, 2007). These three types of problem solving are illustrated in Figure 3.6.

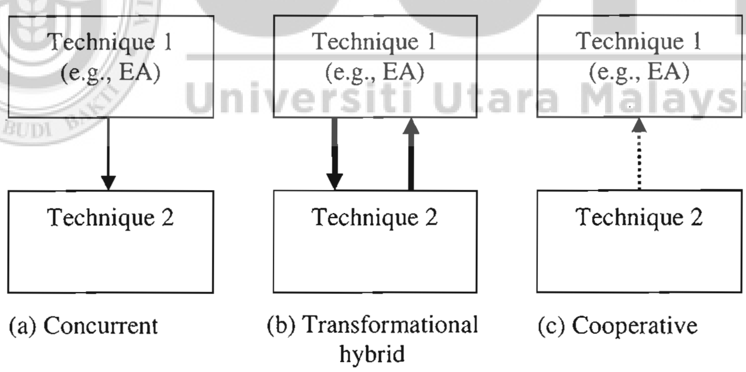


Figure 3.6. Three types of processes in general hybridization architectures

Concurrent architecture involves the whole components of a technique and other technique (which to be hybridized) required for proper functioning in line with the hybridization model, as shown in framework (a) of Figure 3.6. In this architecture, the second technique involves fine-tuning the solution formulated by the first technique, also named as preprocessing method. Here, EA can be the preprocessing method when the fine tuning technique is in a concurrent hybridization model. For instance, evolutionary algorithm is applied at the first stage then followed by a mathematical programming employed at the second stage (Till, Sand, Engell, Emmerich, & Schönemann, 2005; Urselmann, Emmerich, Till, Sand, & Engell, 2007; Tometzki & Engell, 2009). Previous hybridization studies that employed concurrent architecture were such as particle swarm optimization hybrids with geometrical place evolutionary algorithm (Grosan, Abraham & Nicoara, 2005); ant colony optimization with genetic programming (Abraham & Ramos, 2003). In all, concurrent architecture may be the most well-known hybridization architecture.

Transformational hybrid architecture is a circular system involving exchanges of information between two techniques during a problem solving process. This circular process is illustrated as framework (b) in Figure 3.6. EA is fine-tuned by the performance of other technique while the technique is optimizing the performance of EA. This means that both techniques are working in coordination at the same time. For instance, Toledo, da Silva Arantes, De Oliveira and Almada-Lobo (2013) used simulated annealing to intensify the search for solutions in the neighborhood of the best individuals found by the genetic algorithm, while the cavity heuristic determined quickly the values for a relevant decision variable of the processing speed of each machine. Beddoe and Petrovic (2006) used case-based reasoning to store and retrieve constraint violations whilst genetic algorithm improved the accuracy of case-based reasoning and reduced the features of storing amount by selecting and weighting them.

Framework (c) in Figure 3.6 illustrates cooperative architecture. It is a minor hybridizing system where the hybrid technique is not required for proper functioning of the system. In fact, the two techniques are intricately entwined. For instance, part of technique 2 may aid in parameters determination or initialization phase of technique 1. To our knowledge, little is explored on cooperative architecture hybridization. However, Grosan and Abraham (2007) argued that although cooperative architecture is merely a small part of hybridization, it is flexible and simple in incorporating the advantages of different techniques. For that reason, our proposed model is the cooperative model.

3.5 Reasons for Hybrid Evolutionary Algorithm

Based on previous studies discussed in Section 2.7, the hybridization of EA may be a suitable technique to cope with the nurse scheduling and rescheduling problem since the exact methods involve heavy computation while solving a NP-hard problem due to enormous search spaces (Belien & Demeulemeester, 2008; Choy & Cheong, 2012; Clark & Walker, 2011; Maenhout & Vanhoucke, 2008b). Moreover, while meta-heuristics method as evolutionary algorithm alone may be less effective in coping the dynamic and uncertain nature of modern problems, EA hybridization may not have the same problem. As stated by Grosan and Abraham (2007), hybridization of EA has the potential to challenge real world problems which consist of uncertainty, complexity and vagueness. Thus, the hybridization method offers a realistic way of tackling difficult and challenging problems.

In fact, the key advantage of this technique lies in its limitation. EA is capable in searching large search space but it is less effective in identifying local optima

(Whitley, 2001). For the sake of balancing the exploitation and exploration, a hybridization strategy is used to address this drawback. Some well-known cooperation that hybridize EA with local search are memetic algorithm (Martínez-Estudillo, Hervás-Martínez, Martínez-Estudillo, & García-Pedrajas, 2006; Korošec, Bole, & Papa, 2013; Lacomme et al., 2005; Ramli, 2004; Sorensen & Sevaux, 2006; Sudholt, 2009) and differential evolution (Wang & Li, 2010). For example, Wang and Li's (2010) quantum-inspired evolutionary algorithm (QEA) used quantum gate rotation in qubit-based hyperspace to search for good search space direction, but it lacked certain ability for local exploitation. On the other hand, differential evolution (DE) performed evolutionary search by using differential operator and one-to-one competition scheme in space of real value, but it lacked certain mechanisms for global exploration. Therefore, they used DE to cover the exploitation weakness of QEA. As a whole, EA hybridization studies had experienced better balancing between exploration and exploitation. As Tušar and Filipič (2007) claimed pure GA was inferior to the balancing work especially on multi-objective optimization problems.

Essentially, improving and repairing certain functions of EA are the major purpose for forming various EA hybridizations. To improve EA, hybridization has been focusing on initial population (Khaji & Mohammadi, 2014; Rahnamayan, Tizhoosh, & Salama, 2007; Maaranen, Miettinen, & Makela, 2004). In order to increase the convergence speed due to the problem of high computation time, Rahnamayan et al. (2007) employed opposition-based learning to generate initial population. Khaji and Mohammadi (2014) integrated heuristic into initial population to cover many cavities for generating promising search space. They simplified some equations of a complex function in addition to some random individuals. Moreover, for the sake of disregarding convergence speed, quasi-random initial population of Maaranen et al. (2004) can increase the solution quality as well. This means that initializing the population individuals guided by heuristics is an act in favor of searching only towards some feasible regions.

Besides improving initial population, clustering population individuals also seeks to search potential solutions from the population. In the clustering algorithm of Martínez-Estudillo et al. (2006), clustering process was employed by EA and a local search responded to the evolutionary design of neural networks. EA allowed the selection of individuals representing different regions in the search space. Thus, the optimized individuals were more likely to converge towards different local optima. Furthermore, hybrid EA which used a population with fixed features of local optima was a discrete optimization method used to solve permutation optimization problem (Bożejko & Wodecki, 2005; Rogalska, Bożejko, & Hejducki, 2008).

With regards to individual representation, solutions may be represented in an indirect way as decoding algorithm maps any genotype to a corresponding phenotypic solution. In this mapping, the decoder can exploit problem-specific characteristics and thus apply heuristics (Aickelin & Dowsland, 2004; Lin, Gen & Wang, 2009; Moz & Pato, 2007). Lin et al. (2009) employed an extended priority-based encoding hybrid EA by combining local search and fuzzy logic control (FLC) to enhance the search ability of EA. Moz and Pato (2007) hybridized constructive heuristics with genetic algorithm on specific encoding of permutations and permutation-based operators used for each encoding in a sequencing problem. In that, Moz and Pato

(2007) claimed that the fitness function offers great possibilities for hybridization because it can be used as a decoder that decodes the indirect represented genotype into feasible solution.

There are EA hybridization techniques hybridized after the operation of parent selection (Kumar, Tyagi, & Sharma, 2013; Malim & Wessberg, 2010). The parents selected by roulette wheel selection in Kumar et al. (2013) were then used as the initial search point of hill climbing local search. This hybrid genetic and hill climbing algorithm (HGHCA) were used to increase exploitation. Malim and Wessberg (2010) proposed a memetic algorithm for feature selection in volumetric data containing spatially distributed clusters of informative features in neuroscience application. Consequently, making small moves within a defined search area and thus amplifying its search performance were the improvement of hybridization of selection.

Furthermore, a hybridized technique could aid EA's recombination operations (i.e. crossover and mutation) to improve offspring (Sorensen & Sevaux, 2006; Lacomme et al., 2005; Lin, 2010; Mashwani & Salhi, 2012; Prodhon, 2011) or repair infeasible individual (Aickelin & Dowsland, 2000; Bureerat & Sriworamas, 2013; Salcedo-Sanz, 2009; Wu, Yeh, & Lee, 2015; Zhong & Yang, 2004). Mostly, these studies are multi-objective evolutionary algorithm (MOEA). The hybridized techniques were used to exploit the search from the diffusion search of recombination. For example, Mashwani and Salhi (2012) implemented multiple crossover operators to improve evolutionary optimization. They selected the ideal crossover since different crossover operators were suitable for different problems. However, the multi crossovers separation procedure was problematic due to lack of flexibility in tackling restrictions during crossing over in a crossover operator. The hybridization of Lin (2010) was a combination of differential evolution (DE) with real-valued genetic algorithm (RGA) that specifically used the main perturbation of differential vectors and the minor perturbation of mutation. The differential vector perturbation replaced RGA's crossover by taking excellent individuals as the base vectors. Again, this deepened the search as recombination operations were yearning for exploitation. In improving the production of GA's crossover operator, Lacomme et al. (2005) applied local search with a fixed probability on either one of the offsprings whereas the other one was discarded randomly. The hybridized local search with a fixed probability was used to improve the offspring as well as to keep the diversity of next generation. Perhaps, the exploitation act of local search was ineffective since their study needed a fixed probability parameter and median eliminated replacement strategy to escape from achieving premature local optima.

Repair technique should be a simple refinement procedure that works on contemporary solutions to ensure feasible solutions. In recent researches, Wu et al. (2015) presented particle swarm optimization (PSO) with mutation and refinement procedure to address the fairness of average number of nurse for each shifts. The mutation of GA and local search refinement procedure were the repairing and improving actions upon any infeasibility to the constraints. The hybridization of real code population-based incremental learning with differential evolution (RPBIL-DE) was proposed by Bureerat and Sriworamas (2013). The differential evolution was a network repairing technique that used to tackle multi-objective topological design problems. Korošec et al. (2013) employed memetic algorithm with problem-specific reproduction operators and local search to fine-tune infeasible solutions. In all, local search with some simple swapping procedures were used mostly as the repair technique in hybridization. Noted that the repair technique in permutation encoding is needed to avoid infeasibility which caused by the exploration of crossover operator (Salcedo-Sanz, 2009). Hence, the repair technique is meant to be used after the implementation of EA recombination operators (Salcedo-Sanz, 2009; Wu et al., 2015). This is also alluding to a need of restriction for the unbounded and explosive crossing over.

In conclusion, hybridization may be the key to solve practical problems. Grosan and Abraham (2007) and Eiben and Smith (2003) contended that the hybridizations that incorporated evolutionary algorithm have mostly improved the performance of the evolutionary algorithm in terms of convergence speed and the quality of solution produced. In 1990s, local neighborhood search or extensions such as simulated annealing could be imbedded in genetic algorithm to enhance performance (Reeves, 1993). However, the present EA hybridization has not only hybridized with local search series, but also has been expanded to particle swarm optimization (PSO) (Grosan et al., 2005), artificial bee colony (Xiang, Ma, & An, 2014), case based reasoning (CBR) (Beddoe & Petrovic, 2007), ant colony optimization (ACO) (Aickelin, Burke, & Li, 2007) and others. In view to the cooperative architecture perspective that involves a small part of hybridization, meta-heuristics that have simplicity characteristic shall be mainly considered. Thus, our cooperative model focuses on cuckoo search to obtain simplicity.

3.6 Restriction Enzyme

Restriction enzyme is an enzyme that cuts DNA after recognizing a specific sequence of DNA (Pingoud, Alves, & Geiger, 1993; Roberts, 1976; Vincze, Posfai, & Roberts, 2003). Restriction enzymes were named for their ability to limit or restrict the number of strains of bacteriophage (Lederberg & Meselson, 1964). Thus, the biological role of restriction enzyme is to protect cells from foreign DNA.

The DNA sequence can be recognized by a restriction enzyme where these stretches of DNA are named as recognition sequences. Different bacterial species make restriction enzymes that recognize different nucleotide sequences (Kessler & Manta, 1990). Therefore, restriction enzymes contribute to genetic engineering applications (Barrangou et al., 2007; Horvath & Barrangou, 2010; Urnov, Rebar, Holmes, Zhang, & Gregory, 2010). Restriction enzymes rearrange genes to create new combinations of DNA to achieve the goal of genetic engineering, which is changing the genetic makeup of an organism.

In all, restriction enzyme is normally used to identify DNA rearrangement whereby we adopted this concept into recombination strategy of hybrid evolutionary algorithm and cuckoo search. This Restriction Enzyme Point has first been practically used in crossover operator.

3.7 Cuckoo Search Concept

Cuckoo search is inspired by the brood parasitic behavior of some cuckoo species. This nature-inspired metaheuristic was developed by Yang and Deb in 2009. Standard Cuckoo Search (CS) rules are simplified as below (Yang & Deb, 2010):

1) Each cuckoo lays one egg at a time, and dumps it in a randomly chosen nest.
2) The number of available host nests is fixed, and a host can discover an alien egg with a probability $P_a \in [0, 1]$. Once discovered, the host bird can either throw the egg away or abandon the nest so as to build a completely new nest in a new location.
3) The best nest with high quality of eggs (solutions) will carry over to the next generations.

Based on these rules, the implementation of cuckoo search has been transformed to pseudo code as shown in Figure 3.7. In that, Yang and Deb (2009) experienced the random walk via levy flights in cuckoo search. Generally, a random walk is a Markov chain in which the next location should merely depend on the current location and transition probability.


```
Start
    Objective function f(x), x=(x1, ···, xd)^T
    Generate initial population of n host nests xi (i = 1, 2,···, n)
While(t< MaxGeneration)
        Get a cuckoo randomly by performing randomwalk
        Fitness Fi quality evaluation
        Choose a nest among n (say, j) randomly
    If (Fi>Fj)
            Replace j by the new solutions
    End
        A fraction (Pa) of worst nests are abandoned and new ones are built
        Keep the nests with quality solutions
        Rank the solutions and find the current best
End
    Postprocess results and visualization
End
```

Figure 3.7. Pseudo code of Cuckoo Search algorithm

Additionally, two phases of mimicry feature inspired by the natural brood parasitism of cuckoo breeding behavior are technically not involved in cuckoo search algorithm. They are the behaviors before hatch and after hatch. Before the hatching behavior, some female parasitic cuckoo species are specialized in imitating egg color and pattern of host species, whereas host species evolve different strategies as host defenses in order to recognize their own eggs. The interactions between species can be remarkably sophisticated that presents a beautiful example of evolution and adaptation. Practically, mimicry of host egg has not been applied to cuckoo search. This is because a subjective perception of mimicry gives lack of diversification in reproduction strategy which is against the aim of having massive searching approach. However, Yang and Deb (2010) asserted that the intention of this mimicry in color and pattern is to prevent cuckoo eggs from being abandoned so as to increase their reproduction. Furthermore, another significant trick of mimicry feature after hatching behavior is that cuckoo chicks are able to mimic the call of the host chicks (Marichelvam, Prabaharan, & Yang, 2014; Yang & Deb, 2010). The reason for cuckoo chicks to catch attention from the host bird is to gain access for more feeding opportunities. This move may aid the cuckoo chick to stay longer and survive in the communal nest.

3.7.1 Advantage of Cuckoo Search

A good balance of intensive local search strategy and an efficient exploration of the whole search space may usually lead to a more efficient algorithm. Yang and Ded (2010), the creators of cuckoo search, stated that there are several advantages of applying cuckoo search. Previous studies (Brajevic, Tuba, & Bacanin, 2012; Marichelvam et al., 2014; Rodrigues et al., 2013; Yang & Deb, 2010; Yang, 2014) tested that cuckoo search was efficient than particle swarm optimization (PSO) in terms of global convergence requirements because PSO may converge prematurely to a local optimum.

For hybridizing improvement, cuckoo search and genetic algorithm can be compared as follows. Firstly, cuckoo search can be a population-based algorithm that has its global-search ability (Marichelvam et al., 2014). This is quite alike to genetic algorithm. Secondly, the randomization is more efficient because the step length is heavy-tailed, and any large step is possible (Tuba, Subotic, & Stanarevic, 2011). It is also able to have a wide exploration or deep exploitation by controlling the stepsize. Thirdly, a parameter used to be tuned is lesser than genetic algorithm since the latter obtains several operators. Thus, cuckoo search can be adapted to wider optimization problems (Yang & Deb, 2010). Lastly, although each nest is denoted as one solution in the standard cuckoo search (Yang & Deb, 2010), it also can be represented as a set of solutions. To this regard, cuckoo search is easy to be extended to the type of meta-population algorithm. In sum, Brajevic et al. (2012) pointed out that efficiency and simplicity are the superiorities of cuckoo search. Therefore, by considering the advantages, cuckoo search may have the potential to be adapted into a population-based condition of genetic algorithm.

Although cuckoo search is simple, it does not have any matter with problem solving ability. Cuckoo search has addressed several theoretical problems. For instance, traveling salesman problem was solved by discrete cuckoo search (DCS) where its population was restructured (Ouaarab, Ahiod, & Yang, 2014). Besides that, because the parameters of cuckoo search were constant, Marichelvam (2012) developed an improved hybrid cuckoo search (IHCS) to overcome the inefficiency. In his study, permutation flow shop scheduling was solved successfully by the IHCS. Marichelvam et al. (2014) employed improved cuckoo search (ICS) which incorporated constructive heuristic to the initial solutions for solving multistage hybrid flow shop scheduling problem in furniture manufacturing. Additionally, Tuba et al. (2011) modified the random walk with a sorted function in order to solve unconstrained optimization problems. Binary cuckoo search (BCS) measured optimum-path forest classifier was studied by Rodrigues et al. (2013) for the optimization problem of feature selection. Cuckoo search was also applied to address *k*-dimensional optimization problem. It searched the feasibility of maximizing entropy criterion for multilevel image thresholding selection (Brajevic et al., 2012).

For more review on the achievements of cuckoo search, Yang and Deb's (2014) work can be consulted. Although the applications of cuckoo search are rapidly expanding recently, there are rooms for other industrial applications especially to a nurse scheduling problem in healthcare services. After all, cuckoo search is a new search technique that still has great potential.

3.8 Summary

Overall, evolutionary algorithm can produce more than one solution at a time. This is because the technique explores different regions of search space to gradually enhance performance. The search operators are always surrounded by the population's selective pressure, convergence issue, randomization and diversity, which are all committed to balancing exploration and exploitation even though these two principles are contradictory. Therefore, harmonizing or employing a balanced approach is essential in hybridization to achieve effective search.

Furthermore, the origin concept of restriction enzymes can flexibly amend the genetic makeup of an organism to control and recombine another new DNA. On the others hand, efficiency of search and simplicity of implementation are the superiorities of cuckoo search. These aspects may be suitable to complementary the exploitation lack of evolutionary algorithm.

In sum, for solving a complex nurse scheduling and rescheduling problem, a cooperative hybridization is suggested. Evolutionary algorithm is the preprocessing method of choice, the concept of restriction enzyme and cuckoo search component are the cooperative approach. Specifically, the cooperative approaches are incorporated as an enhancement of parent selection and crossover operator performance in order to produce high quality solutions (schedules). The purpose is to improve the overall performance of evolutionary algorithm with flexible attribute when vast numbers of constraints involved. It is envisioned that the hybrid evolutionary algorithm is possible to diversify the search and lessen the high probability of producing an infeasible solution, while at the same time inheriting most properties of their parents.

CHAPTER FOUR

METHODOLOGY OF NURSE SCHEDULING AND RESCHEDULING

Previous chapters discussed the problem of nurse scheduling and rescheduling and identified potential hybrid search techniques. This chapter discusses the research design used to accomplish our research objective to develop an effective nurse scheduling and rescheduling model.

4.1 Research Design

This research is to develop a model for integrating nurse scheduling and rescheduling problem (NSRP) that concurrently considering nurse capacity, preferences, and uncertainty. A nurse schedule with minimum constraint violations is aimed, as similar as in producing a readjusted schedule. Next, this research involves primary data and secondary data. The data were gathered by interviewing the field experts and collecting relevant reports such as rules and regulations of wards, implemented past schedules, and literature review. In that, all constraints are identified whereby develop and formulate our model for NSRP. The model is developed by a cooperative architecture hybridization that based on Evolutionary Algorithm (EA) and Cuckoo Search (CS), in which three new parent selection operators and two new crossovers operators are proposed. Lastly, quantitative analysis is used in our model evaluation and validation. A number comparison experiments carried out on each the proposed technique, which these results altogether were later used to find out the most fitting operators. In models comparison, two classical models are used for benchmarking the proposed models comprising the earlier identified fitting operators, to select the best model. Moreover, *what-if* analysis is employed in rescheduling context and the result validated with Moz and Pato's (2007) retrieval result. Overall, an overview of the research process is illustrated in Figure 4.1.

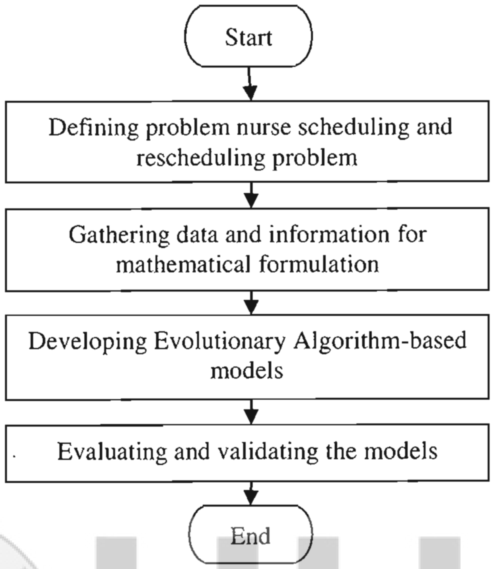


Figure 4.1. Research process

The first three phases are dealt within this chapter whereas the next chapter is devoted to explaining the fourth phase. In order to plan and structure a more thorough research process, the flow of research activities that detail the implementation to attain the specific objectives are shown in Figure 4.2.

Phase	Implementation	Research Questions	Specific Objective
Start			
Phase 1: Problem description	▪ Reviewing the literature ▪ Collecting voice of experts	▪ 'What' are the characteristics of a nurse schedule	Specific Objective 1
Phase 2: Information gathering and mathematical formulation	▪ Collecting data: – Interviewing the experts (e.g., matron, head nurses, nurses) – A yearly schedule of two different wards – Constraints identification ▪ Developing mathematical objective function and constraints for the minimization problem	▪ 'What' are the constraints ▪ 'What' is the methodology needed for model formulation	Specific Objective 1, 2
Phase 3: Models development	▪ Developing nine models through EA operators enhancement and integration of CS to solve NSRP ▪ Using MATLAB language	▪ 'How' is the scheduling and rescheduling models proposed ▪ 'What' operators need to be improved	Specific Objective 3, 4
Phase 4: Model evaluation and validation	▪ Comparing the models (e.g., effectiveness, efficiency, accuracy, reliability) ▪ Comparing the models with two well-known EA models ▪ Using *what-if* analysis in rescheduling context and comparing it with Moz and Pato's (2007) retrieval result	▪ 'How' are the models evaluated ▪ 'What' and 'Why' is the best model selected	Specific Objective 2, 5
Start			

The specific objectives are:

1. To identify all relevant constraints and parameters that makes up all rules and nurses preference as far as possible within appropriate levels of nurse skills and staffing size.
2. To determine the degree of adjustment that will give low impact on other nurses in the rescheduling problem (with regards to schedule disruption, quality retrieval, fair on-call delegation, and high nurse preferences as *Integrated Requested Off* days). Controlling for nurse coverage and high nurse preferences were not only aimed at scheduling but also during rescheduling to obtain a win-win situation in a schedule that complements the nurses' contract and their personal request.
3. To construct new modified parent selection operators to acclimatize population diversification.
4. To construct new modified crossover operator for a scheduling problem and present it as a repair operator for a rescheduling problem by promoting a more flexible way of crossing over and adding little exploitation element to avoid a slow convergence.
5. To evaluate the performance of several evolutionary models and the *what-if* analysis.

Figure 4.2. Details of research activities

4.2 Problem Description

A nurse schedule comprises work-related activities and home-related activities that lead to different nurse preferences. Moreover, the schedule frequently changes due to uncertainties. This research intends to offer a well-planned and well-executed nurse schedule by developing a nurse scheduling model as well as integrating rescheduling components for real-time scheduling instances. Notably, in case of rescheduling, it needs to reproduce a feasible retrieved schedule, in that, the original schedule is not ignored.

Scheduling and rescheduling are mutually exclusive. Controlling for nurse coverage and high nurse preferences are vital in scheduling and rescheduling to obtain a win-win situation in a schedule that complements the nurses' contract and their personal request. Objective 2 stresses some rescheduling aspects.

4.3 Gathering of Information

This research employed two types of data, i.e. primary data via a series of interviews with experts and secondary data. The interviewees were the matron and head nurses at two different wards at Hospital Sultanah Bahiyah (HSB). They talked about their experiences, opinions, preferences and organizational processes. We adopted the questionnaire of Ramli's (2004) and restructured it to suit the current nurse scheduling and rescheduling conditions. Also, secondary data were collected such as schedule records, official publication, and annual reports. As a result, we obtained a yearly record of implemented schedules from the Cardiac Rehabilitation Ward (CRW) and the Emergency Department (ED) ward.

HSB was selected because it was considered a large general hospital whose scheduling problems were typically representative of the problems in other hospitals such as private hospitals or teaching hospitals throughout Malaysia. CRW and ED wards of HSB were studied because both wards behaved differently from each other, hence had different changes in shifts. Data collected showed that nurse demand in CRW could represent a typical ward while the ED ward was more likely to employ a rescheduling strategy than CRW because of uncertain number of outpatients. ED involved impromptu incidents more than routine workload. However, both had common nurse scheduling and rescheduling characteristics. They were considerably large wards in terms of size, but were still facing staff shortages in real-time, had various types of shifts, involved different skills of nurses, and had high frequent shift adjustment. In this respect, our model could mirror a real-time situation more comprehensively and could offer an acceptable benchmark.

4.3.1 Types of Data

Three types of data were collected. They were the number of nurses and their skill level, the duration of schedule in term of days, and shift types. In addition, a number of constraints the nurses faced were collected to develop a quality schedule.

4.3.1.1 Number of Nurse

Each ward had a different total number of nurses. Despite this, the skill level was also an important consideration. Basically, there were a small number of senior nurses and a bigger number of junior nurses in a ward. The senior nurses were assigned evenly to each slot of the ON duty shift. This condition would change once the junior nurses were able to enhance their skill under a senior nurse's supervision.

4.3.1.2 Duration of Schedule

Head nurses were responsible to plan a fortnight schedule, which might change due to unexpected circumstances. In fact, head nurses who plan the schedule manually might face difficulty while making the change. Practically, a schedule could be classified as weekly period and weekend period to cater for the needs of the nurses.

4.3.1.3 Shift Types

The mentioned irregular shifts manner in chapter 2.6.7 has rarely been used in Malaysia's hospitals. It introduces no fixed temporal pattern. Thus, this erratic shift pattern gives lease control of work schedule and may lose the consistence patient care which given by different nurses (Williams, 2008). Therefore, this research implemented several types of shifts within a fixed period of time, especially in Malaysia context. As shown in Table 4.1. The types of shifts considered were ON duty shifts and OFF duty shifts.

Table 4.1

Types of Shifts in Each Duty Shift Classification

Duty shift classification	Types of shifts
ON duty shifts	• Morning shift (i.e. 7 a.m. to 2 p.m.) • Evening shift (i.e. 2 p.m. to 9 p.m.) • Night shift (i.e. 9 p.m. to 7 a.m.)
OFF duty shift	• Weekly day off • Weekend day off • Public holiday leave • Request off day as annual leave • Integrated Request off (i.e. requested weekly off day, requested weekend off day, requested public off day) • Compensation off duty (i.e. Night duty compensation off, and on-call compensation off) • Unexpected leave

ON duty shifts consisted of morning shift which started from 7 a.m. to 2 p.m., evening shift from 2 p.m. to 9 p.m., and night shift from 9 p.m. to 7 a.m.. Each slot of ON duty shift required a certain level of nurse coverage with different skill levels to run the ward. On the other hand, OFF duty shift was more complicated with regards to weekly day off, weekend day off, public holiday leave, requested annual leave, compensation off duty (either for night shift duty or on-call duty), and unexpected leave. Unexpected leave was defined as being not available in the position such as delivering outpatient care, attending training, taking sick leave, and taking personal emergency leave. Each leave had its own rules and regulations.

To get an annual leave, the following list should be met: *who are eligible to apply*, *when (i.e. on which day) the leave is to be taken,* and *how many off days requested.* Regarding the features of disruption, the information needed was the *number of absent nurses*, *the day of absence*, and the *number of consecutive days with absences.* In practice, it is difficult to concurrently fulfil nurse coverage and satisfy a bunch of necessary off-duties within a little period of time. Therefore, our new *Integrated Request Off* concept was an off day authorized by the head nurse which complemented the off day requested by a staff nurse (e.g., a head nurse can intelligently schedule a weekly off day to the day which a nurse requests an off). By doing so, a harmonious working environment could be achieved as a result of consideration from both sides.

4.3.2 Constraints Identification

Based on the data collected, a list of constraints had to be considered to produce a quality schedule. Generally, there are more attentions given to a hard constraint than a soft constraint. Hard constraint regards to the alignment of hospital regulations whereas soft constraint regards to nurse preferences. Nevertheless, the complementary of head nurse' preference (i.e., regulations) and nurses' preference was crucial because the highest quality of schedule could only be attained by the extent to which constraints were fulfilled. In a complex problem like combining scheduling and rescheduling, it was insufficient to use hard and soft constraints only. This research executed three classes of constraint which were hard constraints, semi-hard constraints, and soft constraints (refer to Section 3.3.3.2). In our point of view, semi-hard constraints are important in linking the scheduling phase and the rescheduling phase. Clark and Walker (2011) contended that some typical preferences included in a scheduling phase should not be disregarded in a rescheduling phase. Therefore, we maintained all constraints in the initial scheduling till the rescheduling, so that the rescheduling phase adhered to the same restrictions to develop a quality schedule after disruption. Essentially, a hard constraint is regarded as the root that determines the feasibility of a schedule. A hard constraint is considered at the survival stage. A semi-hard constraint, on the other hand, is considered at the enduring stage that utilizes the available nurses. Lastly, a soft constraint is considered at the ideal and satisfactory stage.

Generally, this research consists of nine types of constraints that discussed in Section 4.3.2.1 until Section 4.3.2.9. In all, the hard constraints of this research were certainly applied to Basic rules, Nurse workload, Skill classification, and Daily adjustment rules. Moreover, Weekend duty was certainly classified as semi-hard constraint. However, the classification of the others four types of constraints (i.e. Nurse coverage, Work stretch, Day off constraint, and Shift ordering constraint) were slightly complicated which involved all three classifications in one fell swoop. Each of the four constraints types have partly fallen into hard, semi-hard, and soft constraints that corresponding to the importance of each specific items or constraint. All these constraints classifications were clearly listed in Section 5.1's Figure 5.1.

4.3.2.1 Basic Rules

Hospital rules and regulations are important to be noted in scheduling as well as in rescheduling. It is a basic requirement of a schedule for assigning nurses to each ON and OFF duty shifts, every day. Necessarily, each nurse only worked one shift per day. This means that overtime was not permitted in our context.

4.3.2.2 Nurse Workload

A nurse workload was defined as a total number of days worked within a week. Nurses were required to work a total of six days a week and given a rest day (6D1F). The hospital regulation was applied to both scheduling and rescheduling phases. Hence, a workload involving more than six consecutive days for each nurse was not allowed in our context.

4.3.2.3 Weekend Duty

In an effort to promote work life balance and supportive work environment, a weekend off duty was necessary. In this context, at least 1 weekend off duty must be given to each nurse in a fortnight. This constraint was considered in our scheduling and rescheduling context. Weekend was defined as day 6^{th} and day 7^{th} of the week. The challenge here was that the head nurse should monitor the nurse coverage while putting in place this constraint at the same time. This is because all nurses were to rotate in taking the 4 days in the weekend in a fortnight schedule (i.e. day 6th, 7th, 13th, and 14th).

4.3.2.4 Skill Classification

Assigning a senior nurse to supervise a junior nurse was considered in the scheduling phase and the rescheduling phase. As a constraint, nurses with higher skill could undertake jobs carried out by lower skilled nurses but not vice versa; a junior nurse was hardly upgraded to a senior nurse. As a result, with fewer number of senior nurses, the head nurse emphasized that at least 1 senior nurse should be on duty in a ward in each working shift. The same condition applied to rescheduling.

4.3.2.5 Nurse Coverage

To meet the need of a ward, covering constraints were considered. There were two aspects of nurse coverage: coverage for each work shift in a day (by column in the schedule), and coverage for each work shift per nurse (by row in the schedule). Both were accounted for in the scheduling and rescheduling phases.

In a day, at least a minimum and maximum percentage of nurse coverage for each ON duty shift (e.g., morning shift, evening shift, and night shift) should be met. Therefore, three stages of nurse coverage were formed: lower bound, ideal coverage, and tolerable coverage (within the lower bound and ideal coverage). Generally, nurse coverage was aimed to plan ideally in scheduling and adjusted accordingly in rescheduling due to uncertain disruption. In fact, the number of available nurses in each day was not static and varied by ward. The nurse coverage's requirement might be adjusted accordingly depending on the number of nurses available each day. For instances, based on the interview, the number of nurses needed for each ON duty shift in CRW and ED is summarized in Table 4.2.

Table 4.2

Daily Nurse Coverage of CRW and ED

	CRW		**ED**	
	Lower bound	**Ideal**	**Lower bound**	**Ideal**
Total No. of Nurse	24	24	34	34
Morning Shift	4	5	5	7
Evening Shift	4	5	5	7
Night Shift	3	4	5	7
Senior Nurse per Shift	≥ 1		≥ 1	
On-call nurse (standby) per Day	1		1	

In terms of nurse on shift, the covering constraints signified fair allocation especially to night shift duty and on-call duty. For each nurse, the night shifts must constitute approximately 25% of their total workload within a fortnight. So that, each nurse must work a night shift per schedule. In other words, there must be at least *n* number of consecutive 'N' shifts assigned to each nurse. The complexity of nurse coverage was increased when seniority involved. Particular for night shift purpose, head nurse would like to distribute the available number of senior nurses evenly and sufficiently. Also, at least one senior was assigned in each ON duty shifts of a day. This was to avoid a condition that has no senior nurse assisted the junior nurses during the working shifts.

In the rescheduling stage, asking the same nurse to fulfil on-call duty while ignoring the others was discouraged. Thus each nurse having an equal chance of being assigned to an on-call duty was considered in the research.

4.3.2.6 Work Stretch

Two aspects were considered to prevent the nurses from experiencing fatigue. These constraints were important to be taken into consideration in the scheduling phase and rescheduling phase. Firstly was with regards to the number of a stretch. A nurse was not allowed to take more than 6 consecutive ON duty shifts. Also, a single work day was discouraged since most nurses preferred consecutive off duty shifts. For that reason, split off days were violated unless the nurse him/herself requested to be off duty on that particular day.

Secondly was about the type of ON duty shifts in a stretch (i.e., a number of consecutive empty cells). In the process of filling a stretch, an approximate equal number of two ON duty shifts (i.e., 'M' and 'E') in a stretch was assigned. This was to ensure that each nurse (i.e., by row) had an equal number of morning shifts 'M' and evening shifts 'E'. Based on the data collected by Ramli (2004), Table 4.3 illustrates the concept of the preferred combinations of morning shift (M) and evening shift (E).

Table 4.3

Work Stretch Concept

Size of work stretch	Work stretch
For 6 days work stretch	A combination of $2 \leq M \leq 4$ and $2 \leq E \leq 4$
For 5 days work stretch	A combination of $2 \leq M \leq 3$ and $2 \leq E \leq 3$
For 4 days work stretch	An equal balance of 2M 2E assignment
For 3 days work stretch	Same type of shift either 3M or 3E
For 2 days work stretch	Same type of shift either 2M or 2E

4.3.2.7 Day Off Constraint

Different types of off duties are preferable to be arranged together in order to construct a consecutive off days. This was giving more resting time to the nurses. Hence, this was not that hard to understand why split off day pattern (0-1-0 case) was not preferred. Besides the mandatory regulation of off day, Day off constraints was mainly focusing on nurse preferences as a retention strategy that fights against turnover crisis. Therefore, the perseverance of approving off duty in a consecutive manner or in a timely manner (e.g., requested off day and compensated off day) has been continually taken up in the rescheduling stage, not merely for scheduling stage.

On the other hand, there were constraints which used to forbid nurse overworked and give timely rest. In this research, a certain number of off duty days was given to the nurse who was assigned to three consecutive night shifts. Besides, off day was most likely to be used to compensate on-call duty, if and only if the selected nurse was originally assigned to off duty shifts. The on-call compensation constraint was stressed in the rescheduling phase.

4.3.2.8 Shift Ordering Constraint

A nurse was not allowed to work two shifts continuously in a day. The order of assigning adjacent ON duty shifts for each nurse generally follows the rule of circadian rhythm, M<E<N. For instances, assigning M followed by E, or E followed by N, or M followed by N. These orderings were to longer a nurse' rest time during his/her shift transition. In this respect, a nurse who worked in a night shift must not be assigned a morning shift the next day. This shall be obeyed even during rescheduling. During the implementation period, shift ordering was even more restricted. As in rescheduling, the previous shift of the day was considered past (*static*); hence only little alteration can be made towards the adjacent shift. Next, any compensation off duty shift could be claimed right after the work was done, with no compromise. In sum, the whole idea was to provide sufficient rest time for nurses before they start a new shift.

4.3.2.9 Daily Adjustment Rules

To overcome uncertainty, rescheduling was carried out cautiously because it involved additional constraints. When a sudden fine tuning had to be done, this had some impact on the original schedule. The impact that rendered the original schedule unpredictable or unreliable was discouraged in our study. For that reason, this research added pre-retrieval before retrieval to avoid such circumstance.

In pre-retrieval, retrieval decision was made in accordance with how worse a disruption could be and how well a schedule was ready. In our context, the readiness of a schedule basically aimed at having an ideal coverage with high nurse preferences. Thus schedule disruption on hard constraint violations and number of manoeuvred nurse on hand are needed to be understood before retrieval was decided. Each nurse may not simply change her shift for a replacement duty unless it was necessary. Disruption that caused an infeasible schedule was prerequisite for the retrieval process. As a result, daily adjustment constraint had to be considered.

In retrieval, a schedule must fine-tuned by retrieval operator when there was still an infeasible disrupted schedule after pre-retrieval. The retrieval abides by the mentioned types of constraints as well as some additional principles that comply with human rights and forward clockwise direction rule. The head nurse was responsibility to be fair in allocating nurse to on-call duty. The on-call delegation should not consider nurses who took unexpected leave on that day. Additionally, rescheduling could restart or tune at any shift period in a day of a week. The replacement of shifts should obey the forward clockwise direction rule. That is, M < M/E/N/Off < E/N/Off < N/Off < Off. This is because when implementing a schedule, past, present and future shifts are identified. For instance, when an evening shift was disrupted because of absenteeism (*present shift*), the gap must not be replaced by a nurse who had worked in the morning shift (*past shift*). In other words, the nurse who had done his/her duty should not be considered. Thus, a backward order was not allowed in shift replacement.

Minimizing the differences in the original schedule and a retrieved schedule is a general aim of rescheduling. Thus, a total number of changed cells was considered and denoted as quantity change in this research. Nevertheless, in order not to overlook quality change as Clark & Walker (2011) concerned; this research considered both quantity and quality changes in the retrieval operator. The quality change was denoted by two types of schedule adjustment. They were non-radical change and radical change in order to produce best retrieval fitness. Non-radical change was re-adjusting the schedule due to disruption that occurred on the day whereas radical change was re-adjusting the whole schedule which radically changed the remaining shifts in the original schedule.

As a conclusion, in all the nine types of constraints, basic rules, nurse workload, skill classification, weekend duty, nurse coverage, work stretch, and shift ordering constraint have been implemented comprehensively in previous studies (refer to Table 2.3 in Section 2.6), also, Van den Bergh et al. (2013) gave a thorough review on that as well. Nevertheless, this research advanced the constraint of daily adjustment rules in rescheduling phase of NSRP due to ensure low impact on other nurses after schedule disrupted and uphold fair on-call delegation (refer to $f_3(s)$ and $f_4(s)$ of objective functions in Equation 4.2). Moreover, the day off constraint was advanced in this research in terms of nurses' requested off days due to fulfil timely nurse preferences. This type of constraint embedded in $f_2(s)$ objective function (refer to Equation 4.2 in Section 4.4).

4.4 Model Formulation

In this research, an optimal schedule was achieved when the smallest number of fitness or no constraint violations occurred. It means that a schedule, s or an individual performance was evaluated by minimizing the penalty function, $F(s)$ that violated hard and soft constraints. Particularly, greater penalty values were given to constraints which were more important. As shown by Soubeiga (2003) and Ramli (2004), the essential objective function is as follows:

$$F(s) = \min \sum_{k=1}^{t} P_k C_k(s) \tag{4.1}$$

In Equation 4.1, P_k is the penalty value (weight) of violated constraint-type k in t kinds of constraints, $C_k(s)$ is the number of violated constraint-type k in schedule s. The purpose was to satisfy the constraints as many as possible. Hence, in our case, a schedule with a high function value was not preferred when compared with the schedule which had a lower function value.

Our research integrated reactive scheduling in a nurse scheduling problem. Hence, our penalty function may consist of two main parts of function structured by several sub-penalty functions. Basically, the first part of $f_1(s)$ and $f_2(s)$ was used in scheduling and rescheduling which stressed quality of shift arrangements and the last part was used in the rescheduling phase by adding $f_3(s)$ and $f_4(s)$ to evaluate the performance. In all, Equation 4.2 represents the sum of all penalties caused by the constraints violation.

$$Min\ F(s) = f_1(s) + f_2(s) + f_3(s) + f_4(s) \tag{4.2}$$

For our objective function in Equation 4.2,

$f_1(s)$: minimize the violation of nurse work regulations with the number of nurses assigned to each shift in each skill level

$f_2(s)$: minimize nurse's dislike of the shift arrangements to fulfil nurse preferences

$f_3(s)$: minimize the deviation of shift changes after reactive scheduling to ensure low impact on other nurses

$f_4(s)$: minimize the deviation of the same nurse assigned to on-call duty to keep fair on-call delegation during rescheduling

4.5 Evolutionary Algorithm Modelling for NSRP

Our study proposed a hybridization technique to solve a nurse scheduling and rescheduling problem. Evolutionary algorithm is a suitable pre-processing method of choice. Specifically, our proposed technique attempted to enhance the parent selection operator and crossover operator performance in order to provide a better solution. MATLAB R2010a language with Intel® Core™ i5-2410M CPU @ 2.30 GHz and RAM 8GB was used to program the hybrid evolutionary algorithm. The detail of EA modelling is explained in the following sections.

4.5.1 Representation Structure

An individual or chromosome in NSRP was defined as a schedule that consists of three main elements. They are *Nurses (senior and junior)*, *Days (weekly days and weekend)*, and *types of shift.* The chromosome representation is devoted to non-binary matrix representation. Here, a general structure of a schedule (i.e. Individual) is presented in Figure 4.3.

Nurses		Weekly Days					Weekend		Weekly Days					Weekend	
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
senior	1	E	U	E	E	N	N	N	O	O	P	M	M	W	N
	2	N	O	O	P	M	M	Q	N	N	N	O	O	E	U
	3	B	N	N	N	O	O	W	E	T	M	N	N	N	O
Junior	4	N	N	N	O	O	U	M	M	E	E	P	M	M	W
	5	U	M	M	N	N	N	O	O	M	M	E	E	Q	E
	6	M	L	E	E	B	W	N	N	N	O	O	U	M	M
	7	M	M	E	E	P	W	M	E	E	N	N	N	O	O

Shift Symbol	Types of Shift		
M	Morning Shift (7 a.m. – 2 p.m.)	**T**	Requested weekly off
E	Evening Shift (2 p.m. – 9 p.m.)	**Q**	Requested weekend off
N	Night Shift (9 p.m. – 7 a.m.)	**B**	Requested public off
U	Weekly off	**O**	Night shift's compensation off
W	Weekend off	**C**	On-call's compensation off
P	Public off	**L**	Unexpected leave
R	Request off		

* Given that *Requested weekly off*, *Requested weekend off*, and *Requested public off* are in a group of *Integrated Requested Off*

Figure 4.3. A simple structure of a schedule

4.5.2 Generate an Initial Solution

By considering some restricted conditions of scheduling, partial random initial population was activated in a sequence by inserting some basic types of ON duty and OFF duty shifts, in a row-by-row manner. The sequence is listed in Figure 4.4.

1) Generating an initial individual
   i. Inputting night shift 'N' and its night compensation off 'O' in each row, named as *NStretch*
   ii. Entering a list of requested off duty 'R'
   iii. Randomly entering a weekend off 'W' for each nurse in the weekend columns, which are in column 6^{th}, 7^{th}, 13^{th}, and 14^{th}
   iv. Randomly entering a weekly off 'U' for each nurse in the 14 days schedule
   v. Randomly entering a public off 'P' in the weekly or weekend columns
   vi. Randomly entering a stretch of morning shift 'M' and evening shift 'E' into the remaining null cells of a row
2) Repeat step 1 until a number of individuals is created

Figure 4.4. Steps of generating initial population

Population size is a number of individuals in a population. By repeating a specific number of times to generate individuals, an initial population was formed. In this research, the population size of 12 is found to be the most potential size for better performance for the overall models. The experiment is discussed in Section 5.2.

i. Allocation of Night Shift

With regards to *NStretch*, nurses worked a schedule involving three consecutive night shifts and were then given two off days as compensation in the order of { 'N' 'N' 'N' 'O' 'O'}. This arrangement was to provide sufficient rest time for nurses to revert back to a normal sleep cycle. In this study, we started the first *NStretch* randomly in each staff level (Senior and Junior). A staggered pattern was formed after a number of *NStretch* was continually allocated. If *NStretch* rolled to the end row (nurse), the next *NStretch* must be given at the start of the same nurse level, which was then repeated until the end of column. Figure 4.5 shows how *NStretch* was inserted to an individual.

Nurses		Weekly Days					Weekend		Weekly Days					Weekend	
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
senior	1													N	N
	2	N	N	N	O	O									
	3				N	N	N	O	O						
	4							N	N	N	O	O			
	5										N	N	N	O	O
Junior	6	N	N	N	O	O									
	7				N	N	N	O	O						
	8							N	N	N	O	O			
	9										N	N	N	O	O

Figure 4.5. A sample of night shift allocation

ii. Allocation of Requested Off

The allocation of Requested Off was based on a list of Requested Off 'R' planned by the nurses themselves. At first, the nurses were free to apply as many 'R' as needed without exceeding the allocated annual leave. Here, a list of 'R' for each row (nurse) was randomly set as RO[$Nurse_i$, $StartDay_j$:$EndDay_j$] for modal testing. For example, RO[3, 11:12] denotes that nurse 3 had requested 2 consecutive off days from day 11^{th} till day 12^{th} .

iii. Allocation of Weekend Off

For each nurse, a Weekend Off 'W' was randomly placed into column 6^{th}, column 7^{th}, column 13^{th} or column 14^{th}. However, a Request Off shall be replaced by Weekend Off signified as 'Q' in a condition where the planned Request Off was coincidently dropped from the weekend columns. If there was more than one Request Off dropped from the weekend columns, the replacement was randomly picked. The reason for such arrangement was to devise a tolerable schedule that considered both parties' (head nurse and staff nurses) need and preference.

iv. Allocation of Weekly Off

For each nurse, Weekly Off 'U' was randomly placed in a null cell of the schedule. However, a Requested Weekly Off 'T' was given priory to be inserted if and only if the row (nurse) consisted of a planned Request Off. Given that this was used to encourage the head nurse to boost up timely nurse preferences.

v. Allocation of Public Off

By giving a Public Off to each nurse, a condition where the total number of Request Off has more than or equal to one ($R \geq 1$) shall require the Public Off 'P' to be inserted into either one of the planned Request Off cell, as Requested Public Off 'B'. Due to some coverage restriction in a given day, not every nurse was able to take off days on the exact public holiday. Thus, we encouraged the given Public Off to be allocated into a null cell placed next with any one of the off shifts. This arrangement was to increase nurse preference in taking longer consecutive off days. It was an ideal option unless there was no null cell placed aside. In that case, Public Off shall be allocated randomly into a null cell of the row.

vi. Allocation of Morning Shift and Evening Shift

First, we identified the size of consecutive empty cells located in between any 2 Off shifts. Then, we randomly chose a Morning and Evening work stretch pattern to be inserted according to the identified size. We continually inserted the Morning and Evening work stretch till all null cells of rows and columns were filled. Table 4.4 shows the choices of Morning and Evening work stretch pattern for a particular size in a null cell. This list of stretch was employed in the scheduling and rescheduling phases.

Table 4.4

List of M E Stretch Regards to the Size of A Consecutive Null Cells

Size of a null cell stretch	Morning and Evening work stretch
1	M, E
2	MM, EE
3	MMM, EEE
4	MMMM, MMEE, EEEE, EEMM
5	MMMMM, MMMEE, MMEEE, EEEEE, EEEMM, EEMMM
6	MMMMMM, MMMMEE, MMMEEE, MMEEEE, EEEEEE, EEEEMM, EEEMMM, EEMMMM
7	MMMMMEE, MMMMEEE, MMMEEEE, MMEEEEE, EEEEEMM, EEEEMMM, EEEMMMM, EEMMMMM
8	MMMMMMEE, MMMMMEEE, MMMMEEEE, MMMEEEEE, MMEEEEEE, EEEEEEMM, EEEEEMMM, EEEEMMMM, EEEMMMMM, EEMMMMMM
9	MMMMMMMEE, MMMMMMEEE, MMMMMEEEE, MMMMEEEEE, MMMEEEEEE, MMEEEEEEE, EEEEEEEMM, EEEEEEMMM, EEEEEMMMM, EEEEMMMMM, EEEMMMMMM, EEMMMMMMM
10	MMMMMEEEEE, EEEEEMMMMM

Finally, we accepted all surviving (feasible) solutions and non-surviving (infeasible) solutions in the initial population. The non-surviving solution might be capable of initializing a new direction of exploration and escape from being stuck at a local optimum.

4.5.3 Fitness Evaluation

Determining a representative penalty value with the different spectrums of constraints is inevitably desired. As discussed in Chapter 3, we concluded that three classifications of constraints are needed in NSRP. Here, we adopted Maslow's Hierarchy of Needs' principle that constitutes the constraints by understanding their objective and importance, and thus setting a penalty value to achieve quality schedule, as shown in Figure 4.6. The objective function was formulated as in Equation 4.3.

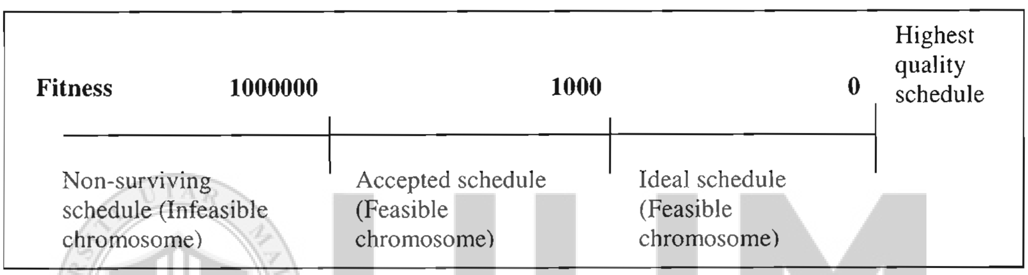


Figure 4.6. Structure of fitness values toward quality schedule

Altogether, we segmented the constraints into 3 main classes by setting up 3 ranges of penalty value to achieve a quality schedule. Based on the importance of each class of constraints, there are at least 1000000 penalty values weighted for a hard constraint, at least 1000 penalty values weighted for a semi-hard constraint and lastly at least 1 penalty value weighted for a soft constraint. Soft constraints by themselves never achieved 0 fitness value because achieving even 1 fitness value was practically impossible (Kelemen et al., 2005). At this point, "how great" a quality schedule may respond to fitness evaluation relied on constraints violation in each class.

The advantage of this computation was that it could aid tracing and identifying a quality schedule. For example, despite a larger value showing larger violation, a fitness of 2005034 indicates that the schedule violated *two* hard constraints that caused infeasibility, violated *five* semi-hard constraints and at most *thirty-four marks* of penalty due to violating soft constraints. Therefore, we can easily trace a schedule problem as well as comparing schedules, by selecting the one that has the minimum fitness value.

Decision variables:

$$X_{vipj} = \begin{cases} 1 & \text{If nurse } i \text{ of skill level } v \text{ is assigned to shift } p \text{ in day } j \\ 0 & \text{otherwise} \end{cases}$$

$$D_{ijp} = \begin{cases} 1 & \text{If a scheduled duty shift } p \text{ of nurse } i \text{ in day } j \text{ is changed} \\ 0 & \text{otherwise} \end{cases}$$

$$C_{vjk} = \begin{cases} 1 & \text{If constraint type } k \text{ for skill level } v \text{ in each day } j \text{ is violated} \\ 0 & \text{otherwise} \end{cases}$$

$$C_{ik} = \begin{cases} 1 & \text{If constraint type } k \text{ for each nurse } i \text{ is violated} \\ 0 & \text{otherwise} \end{cases}$$

Notations:

I = number of nurse i

V= number of skill levels v

J = number of days j in scheduling period

P= number of possible shifts patterns p

K= number of constraint types k

W_{ijp}= weightage or penalty cost for the relative constraints D_{ijp}

W_{vjk}= weightage or penalty cost for the relative constraints C_{vjk}

W_{ik} = weightage or penalty cost for the relative constraints C_{ik}

$$Z_s = \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{v=1}^{V}\sum_{p=1}^{P} W_{ijp} D_{ijp} X_{vipj} + \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{v=1}^{V}\sum_{p=1}^{P}\sum_{k=1}^{K} W_{vjk} C_{vjk} X_{vipj} + \sum_{i=1}^{I}\sum_{j=1}^{J}\sum_{v=1}^{V}\sum_{p=1}^{P}\sum_{k=1}^{K} W_{ik} C_{ik} X_{vipj} \quad (4.3)$$

4.5.4 Parent Selection

For the sake of acclimatizing different levels of population size, preserving the population diversity for exploring and exploiting its solution space was our focus. Besides keeping some dissimilar individuals in the population, our proposed mating strategies involved two common factors: selection pressure of distance measured in the search space, and complementary characteristic of a selected pair that involved elite behavior to promising solution space. In other words, *how* close the relationship between the selected pair, and *who* to be selected were important considerations to produce better offspring for the next generation.

To understand further, we experimented with five parent selection operators in EA. Besides Binary Tournament parent selection and Rank-based parent selection, three were our new proposed parent selection operators called Maximax and Maximin parent selection (MM), Discovery Rate parent selection (D_r), and Discovery Rate Tournament parent selection (D_rT). They were the expansion of the elite selection behavior of Tournament parent selection and Rank-based parent selection because they were developed based on the competing principle. Indeed, superior genes are likely to construct more endurable new generation to the nature (Zhong, Hu, Gu, & Zhang, 2005). Thus, in order to execute an in-depth analysis of the elite selections, these 2 classical operators (i.e., Binary Tournament and Rank-based parent selections) were chosen as the benchmark to verify the performance of three newly modified parent selection operators (i.e., MM, D_r, and D_rT parent selections).

4.5.4.1 Maximax and Maximin Parent Selection

Maximax is defined as maximizing the maximum outcome from every alternative. It was known as an 'optimistic' decision criterion. On the other hand, maximin is a notion of finding an alternative that maximizes the minimum outcome from every alternative, known as a 'pessimistic' decision criterion. Maximax and Maximin parent selection (MM) was inspired by the concept of maximax and maximin which was a decision making approach under uncertainty (Heizer & Render, 2006). MM was suitable to be applied in an uncertain population pool with varying characteristic of individuals. This implied that no filtering operator was needed in generating initial population.

Overall, the centre of attention in this operator was on better fit individual without overlooking any type of group that was either in the best fit group or worst fit group. In this sense, a slight distance between the complementary parents was ensured to uphold diversity in search. Figure 4.7 shows the procedure of MM parent selection in detail.

i. Randomly group half of the total number of individuals from the population into a sub-population
ii. Rank amongst the sub-population
iii. Classify the sub-population into two outcomes which are individuals who have better fitness (maximum outcome) and individuals who have poorer fitness (minimum outcome)
iv. The best (lowest fitness value) of the two outcome groups is then selected
v. The two chosen individuals are then defined as Parent1 (Maximax) and Parent2 (Maximin)

Figure 4.7. Procedure of Maximax and Maximin parent selection

4.5.4.2 Discovery Rate Parent Selection

Discovery rate is a probability pa∈ [0, 1] of host bird used to discover alien egg so as to build a completely new nest in a new location. The host bird generates a new individual once a suspicious cuckoo egg is discovered while referring to a certain probability (Random>Discovery rate). Thus, discovery rate parent selection (Dr) is injecting relative difference into discovery concept in order to determine who shall be paired to proceed to the next recombination strategy. This operator was adapted from Yang and Deb (2010). Through the alien egg discovery process, discovery rate parent selection (Dr) was able to study the impact of dissimilarity between selected parents.

With regards to premature convergence that generally pressures elite behavior, a controllable selection pressure mechanism is suitable. To our knowledge, the difference between selected parents has been ignored in parent selection study although it might be an important key to provide a diverse permutation space for reproduction operators. Therefore, D_r parent selection was proposed. The detail is shown in Figure 4.8.

i. Randomly select two individuals from a population
ii. Set a probability rate of discovery P_d and calculate the Relative difference (RD), where the reference number falls on the individual who has a bigger fitness value

$$RelativeDifference(x, y) = \frac{|x-y|}{\max(|x|,|y|)} \quad (4.4)$$

iii. Discovery verification by comparing RD with P_d.
Discovered if RD $\geq P_d$, accept the two selected individuals as parents1 and parent2 to next recombination purpose
Not discovered if RD $\leq P_d$, return the two individuals into the population and repeat (i) to (iii).

Figure 4.8. Procedure of Discovery Rate parent selection

4.5.4.3 Discovery Rate Tournament Parent Selection

Besides the dissimilarity between the selected parents described in D_r parent selection (Yang & Deb, 2010), an additional focus of the elite element was integrated into this selection intensity. Hence, discovery rate tournament parent selection (D_rT) is a selection operator which consists of tournament and discovery rate characteristics (as disccussed in Section 4.5.4.2). This integration was inspired by tournament parent selection that most likely gives reliable performance upon a promising solution.

The tournament was used to select better fit individual as potential parents. The dissimilarity concept gives a practical advantage to D_r and D_rT parent selections. Both acclimatize to a population's condition which regardless of the whole population's diversity whereby may reduce some heavy computation. In sum, the similarity procedure of D_r parent selection can be referred to Section 4.5.4.2. The detailed procedure of D_rT Parent Selection is shown in Figure 4.9.

i. Randomly select two individuals from a population
ii. Set a probability rate of discovery P_d and calculate the Relative difference (RD), where the reference number falls on the individual who has bigger fitness value
iii. Discovery verification by comparing RD with P_d.
Discovered if RD $\geq P_d$, proceed to (iv)
Not discovered if RD $\leq P_d$, return the two individuals into the population and repeat (i) to (iii)
iv. In a tournament among the two, select the lower fit individual as parent 1. The other is sent back to the population
v. Repeat (i) to (iv) to get parent 2

Figure 4.9. Procedure of Discovery Rate Tournament parent selection

4.5.4.4 Rank-based Parent Selection

A predominantly intensification is stimulated in this rank-based selection scheme as suggested by Maenhout and Vanhoucke (2011). Rank-based parent selection (Rk) is a selection based on position in the individuals rank. Individuals in the population were sorted according to their fitness. Then two individuals were in a highly ranked position were selected. Generally, rank-based selection can maintain a constant pressure in the evolutionary search where it introduces a uniform scaling across the population. Ranking does not influence super-individuals or the spreading of fitness values. Due to elite element involved, rank-based parent selection was implemented for our model validation purpose. Figure 4.10 shows the procedure of rank-based selection.

i. Rank individual in the population according to their fitness ii. Select two high ranked individuals (elite behavior) as parent1 and parent2

Figure 4.10. Procedure of Rank-based parent selection

4.5.4.5 Binary Tournament Parent Selection

Binary tournament parent selection (T) is a tourney of two or above randomly selected individuals from a population. The implementation of binary tournament selection is simple and does not involve fitness sorting. Although all individuals have the chance to be selected in order to preserve diversity, the tournament activity has its bias toward elite individual. In fact, tournament may influence the convergence speed. Thus, tournament parent selection adopted from Burke and Smith (2000) was used to validate the performance of the three newly proposed techniques. Figure 4.11 shows the procedure of binary tournament selection in detail.

i. Randomly pick two individuals from a population
ii. Compare them against their fitness value
iii. The fitter (lower fitness value) is selected as parent
iv. Return the worse fit individual back to the population
v. Repeat (i) - (iv) to get the second better parent

Figure 4.11. Procedure of Binary Tournament parent selection

4.5.5 Crossover Operator

The definition of various matrix crossovers and their drawbacks are discussed in Section 3.3.5. By understanding their limitations, Two-factor Blockwise crossover and Cuckoo Search Restriction Enzyme Point crossover were created in this research to ensure their suitability to our nurse scheduling and rescheduling context. Based on the promising performance of the crossover occurrence rate (Cr = 1) (Adamopoulos, Harman & Hierons, 2004; Eskandari & Geiger, 2008; Kelemci & Uyar, 2007; Miki, Hiroyasu, Yoshida, & Ohmukai, 2000; Srinivas & Patnail, 1994), we adopted Cr = 1 in our context that provides us (1) a strong elitist setting that always input offspring that inherited from elite parents to next iteration by cooperating with parent selection operators; (2) not to ignore a large number of infeasible solutions in the initial population. With a constant occurrence of crossover, the search occupied surviving fitness individuals and non-surviving fitness individuals and thus spread the search space for the region that contains a global optimum; (3) a compulsory fine-tuning. The proposed Cuckoo Search Restriction Enzyme Point crossover obtains directed elements for fine-tuning purpose; (4) an in-depth investigation on the randomization strategy which merely look into the permutation strategy of each proposed crossover operators.

4.5.5.1 Single Point Crossover

Crossover attempts to explore a search space by exchanging genetic information and recombining it from selected parents. In the process of crossing over, randomization is a vital challenge for a complex constraints handling problem. As in our NSRP, randomly change a cell may violate some constraints. For this reason, row-wise crossover of Ramli (2004) is basically upholding some sub-solutions away from excessive disturbance. Basically, row-wise crossover is also analogous to a horizontal single-point matrix crossover. Therefore this single point crossover was adopted and applied to validate the performance of the proposed crossover operators.

Generally, a schedule (chromosome) consists of two parts due to the differences in nurse skill levels (senior group of nurse and junior group of nurse). A detailed procedure of a single point crossover operator is illustrated in Figure 4.12.

i.	Segregate nurses in groups based on their skill levels
ii.	Put a fixed cross point in between the group of senior nurse and the group of junior nurse to identify two blocks of rows
iii.	Swap the block of junior group between the two selected parents. Hence, two complete children are formed
iv.	Calculate the fitness of child1 and child 2
v.	Update current fitness value

Figure 4.12. Procedure of Row-wise crossover

4.5.5.2 Two-factor Blockwise Crossover

Two-factor Blockwise crossover (2Fblockwise) is a matrix form crossover operator which readjusting the chromosome by predetermined blocks. The blocks were cut based on two factors (i.e., horizontal factor and vertical factor). In this research, the classical single point crossover was adopted. To increase the conservative disruption of row-wise crossover from Ramli (2004), 2Fblockwise crossover enhances exploration without excessively messing up some promising sub-solutions.

Figure 4.13 indicates 2Fblockwise crossover with regards to two factors in a two-dimension schedule, which was set apart by one horizontal cross point and vertical cross point. The two factors were the segregation of nurses based on skill levels (e.g., senior group and junior group) and segregation of days based on number of weeks (e.g., 1st week and 2nd week). Therefore, the whole schedule consisted of four (e.g., 2x2) big blocks where the 2nd week of senior group and 1st week of junior group were both swapped in between two selected parents throughout the whole generations.

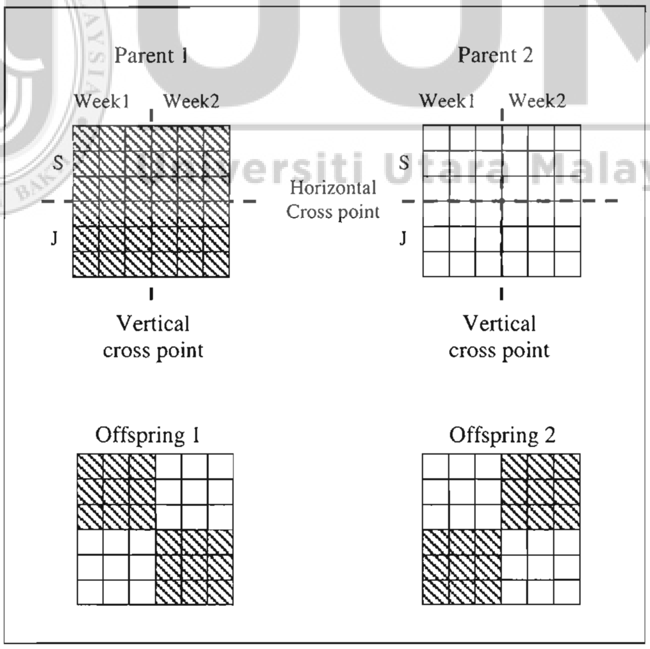


Figure 4.13. 2Fblockwise crossover

4.5.5.3 Cuckoo Search Restriction Enzyme Point Crossover

Basically, Cuckoo Search Restriction Enzyme Point crossover (Max[4x4]CSREP) is defined as a matrix form crossover operator that crossing over various patterns of blocks or sectors between the selected chromosomes to search a better fit chromosome. From the work of Yang and Deb (2010), the acts of generating egg by randomwalk in cuckoo search and a nature of cuckoo mimicry behaviour are adopted in the crossover. Indeed, these acts are intended for searching suitable blocks of crossing over. Figure 4.14 illustrates a simple concept of Max[4x4]CSREP crossover.

In order to reduce unnecessary disruptions, restriction enzyme point (REP) participates in a randomwalk process to crossing over flexibly (see Equation 4.5). As shown in Figure 4.14, REP has no predetermined blocks for crossing over but it leads a cut accordingly at any cell. In our NSRP, REP is a cut after a number of cells are recognized in accordance with hard constraint violation. Based on that, a block is formed by the particular hard violated cell and a group of cells, in other words unreserved shifts patterns. The unreserved shifts patterns involve ON duty shifts and OFF duty shifts, such as morning shift, evening shift, night shift, weekly off shift, weekend off shift, and public off shift.

Futhermore, stepsize is set by random number that relates to the scale of problem of interest (Yang & Deb, 2010). Once a size of block is formed, an imitating process is started by mimicry feature. The mimicry feature is mainly implemented through the shift pattern and blocksize pattern. Again, REP leads the foremost cell according to the unreservedly shifts. Moreover, a size of maximum [4row x 4column] block is compounded by various unreservedly shifts that are joined together at first.

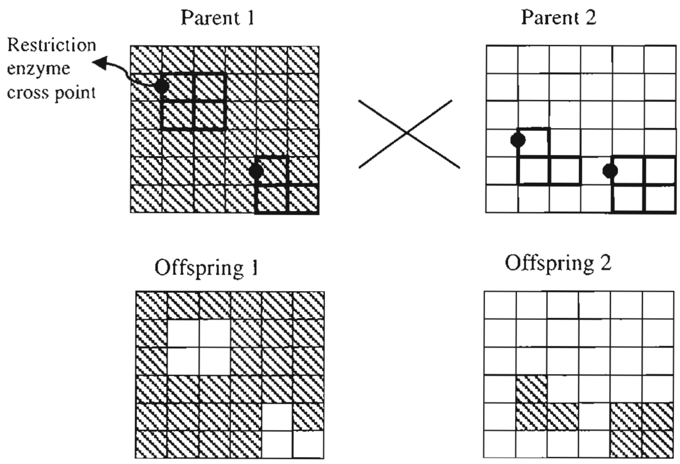


Figure 4.14. A simple concept of CSREP

Essentially, the aim of hybridizing CS is to generate the best child of each parent. For exploration purpose, CS is slightly modified to observe the performance of a child generated from different types of breeding behaviour. Therefore, nearly all Cuckoo Search is hybridized in EA's crossover apart from the discovery rate since Yang and Deb (2010) indicated that this parameter was insensitive to convergence rate. A detailed procedure of this crossover operator and the CSREP$_r$ retrieval operator are illustrated in Figure 4.15.

i. Scrutinize parent 1 in relation to hard constraints violation,
ii. Randomly generate a cuckoo egg of the parent by performing *RandomWalk*
iii. Calculate *RandomWalk* with mimicry feature {shift pattern, block pattern}

$$x_j^{(t+1)} = x_i^{(t)} + S \qquad (4.5)$$

Where $X_i^{(t)}$= vector of P1
$X_j^{(t+1)}$ = vector of P2
S = Stepsize in a random manner
iv. Identify cuckoo egg of P1 with mimicry host egg feature
v. Verify the host egg of P2 by mimicry feature:
   a. Similar bock pattern of P1and P2 is met
   AND
   b. Permissible shift pattern of P1and P2 by restriction enzyme point is met
vi. Repeat (ii) to (v), if (v) if no success
vii. Swap between host egg of P2 and cuckoo egg of P1
viii. Repeat (ii) to (vii) until a fixed number of restriction enzyme cross points are met. In that, a complete cuckoo chick with attentive care and host chick are formed
ix. Calculate and compare the fitness of cuckoo chick and host chick (F_c vs. F_h)
x. Keep the better chick as child derived from Parent 1
xi. Scrutinize parent 2 to repeat the same process from (i) to (x) where now parent 2 is denoted as P1
xii. Keep the better chick as child derived from Parent 2
xiii. Repeat (i) to (xii) until a fixed number of generation is met
xiv. Rank and find the current best-so-far child derived from Parent 1 and Parent 2. The best child derived from Parent 1 is denoted as child 1 and the best child derived from Parent 2 as child 2

Figure 4.15. Procedure of Max[4x4]CSREP crossover

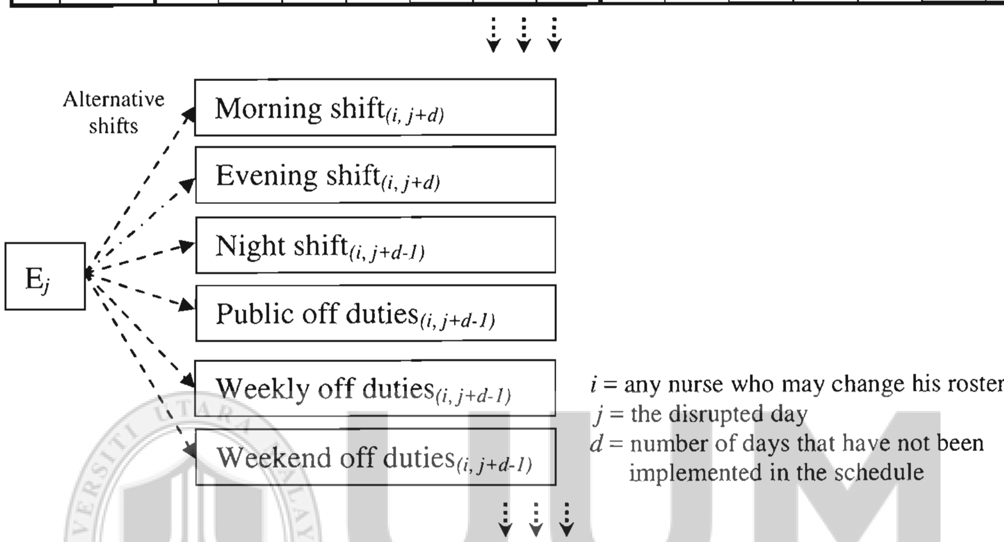
4.5.5.4 Cuckoo Search Restriction Enzyme Point Retrieval Operator

To the extent of Max[4x4]CSREP crossover, Cuckoo Search Restriction Enzyme Point Retrieval ($CSREP_r$) was employed for fine tuning purpose in addressing the rescheduling problem. $CSREP_r$ that engaged with restriction enzyme point and mimicry approach involves evolution and adaptation. $CSREP_r$ is the repair function in rescheduling that should equip with flexibility, sophisticate change and efficiency. In that, having some parameters changed in Max[4x4]CSREP may be pointed to certain level of flexibility that befits retrieval condition.

In $CSREP_r$, three parts were altered from Max[4x4]CSREP Crossover. Based on the Figure 4.15, stopping criteria (refer to Section 4.5.8), unreserved shift pattern of REP (refer to Section 4.3.2.9) and stepsize which depend on the disruption conditions (i.e., number of days absent, number of absent nurse, and number of consecutive days with absences) shall be amended for repairing purpose in rescheduling. This is as complex as searching in a smaller search space that exploration is restricted. In short, a feasible solution with less possible changes made is the vital concern during rescheduling problem.

For instance, with regard to the parameter of unreserved shift pattern in REP, a principle of cyclical rhythm ought to be obeyed during rescheduling. Figure 4.16 illustrates a simple example of shift adjustment in rescheduling phase. Given that the disruption condition is Junior nurse 4 take 1 emergency leave on 9th June 2015. As for a day when the evening shift had a lack of nurses, the scarcity shall mend by night shift nurse or off duty nurse but not by morning shift nurse of the day (see the alternative shifts in Figure 4.16). This is because the morning nurse had already completed her duty. On the other hands, simple adjustment is considered in a condition where the ward operation of the absent day is not disrupted (e.g., no hard constraints violated) such as readjust the absent nurse' schedule with any of his/her ON or OFF shifts in others day, or no shift adjustment will do for other nurses. This means that the absent nurse may take his own responsibility for his/her unexpected leave. Therefore, the well thought-out for repairing cell in rescheduling increases the complexity.

Scheduling		Weekly Days					Weekend		Weekly Days					Weekend	
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
Senior	1	E	U	E	E	N	N	N	O	O	P	M	M	W	N
	2	N	O	O	P	M	M	Q	N	N	N	O	O	E	U
	3	B	N	N	N	O	O	W	E	T	M	N	N	N	O
Junior	4	N	N	N	O	O	U	M	M	**E**	E	P	M	M	W
	5	U	M	M	N	N	N	O	O	M	M	E	E	Q	E
	6	M	L	E	E	B	W	N	N	N	O	O	U	M	M
	7	M	M	E	E	P	W	M	E	E	N	N	N	O	O



Reactive scheduling		Weekly Days					Weekend		Weekly Days					Weekend	
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
Senior	1	E	U	E	E	N	N	N	O	O	P	M	M	W	N
	2	N	O	O	P	M	M	Q	N	N	N	O	O	E	U
	3	B	N	N	N	O	O	W	E	T	M	N	N	N	O
Junior	4	N	N	N	O	O	U	M	M	**L**	E	P	M	M	W
	5	U	M	M	N	N	N	O	O	M	M	E	E	Q	E
	6	M	L	E	E	B	W	N	N	**E**	O	O	U	**C**	M
	7	M	M	E	E	P	W	M	E	E	N	N	N	O	O

Figure 4.16. An example of shift adjustment in rescheduling

4.5.6 Mutation Operator

Generally, mutation operator is used to randomly bit-flipping the selected cell. However, the randomness may easily violate the hard constraints which cause an infeasible solution. Therefore, directed mutation operator is employed to perform little tuning capabilities for the intricacy of highly constraint-based nurse scheduling problem (NSP). Hence, mutation rate has neither not a stressed matter nor pre-determined in the operator. Directed mutations used by Ramli (2004) were adopted and slightly modified with delete-and-insert strategy in this research. Here, the conventional swapping strategy and new delete-and-insert strategy were applied to each mutating operation. Swapping strategy exchanged the position of two selected cells, whereas delete-and-insert strategy deleted the content of a selected cell by inserting a new content into the cell. In addition, 2-dimensional considerations (i.e., columns and rows) were concurrently taken into account in some mutating operations. In all, a sequence of directed mutations was applied to this research. There are called OffTol Mutation, 0-1-0 Mutation, MEBalance Mutation, EN Mutation, NM Mutation, ReqFit Mutation, and WU Mutation.

OffTol Mutation deleted weekend off and weekly off that were not within the weekend area and weekday area, respectively. Here, the operator aimed at assigning equally the number of weekend off duty, weekly off duty, and public off duty for each nurse. The four remaining mutations are 0-1-0 mutation, MEBalance mutation, EN mutation, and NM mutation. They aimed to disrupt the chromosome by row but yet partially guided to avoid further constraint violations. Briefly, in 0-1-0 mutation, a 14-day schedule avoided a single ON duty shift by swapping the only ON duty shift with the next OFF duty shift. Thirdly, MEBalance mutation reduced the imbalance of the total of morning shift and evening shift for each nurse. This operator randomly replaced M with E if the total number of M was lesser than E in a row and vice versa.

Next, EN mutation operator disallowed night shift to be assigned after an evening shift. This mutation randomly swapped E with any M of the row. Fifth, the head nurse was not allowed to assign a morning shift to the nurse who had just finished his/her night shift duty. Hence, NM mutation operator swapped the order N and M in a row with each other.

Sixth, ReqFit mutation aimed to disrupt the M and E gene in a column in consideration of ward coverage daily. It randomly replaced an M with E if the column (per day) had less evening shift coverage and vice versa. OFF duty as 'RO' shall be replaced with the lack coverage of ON duty shift, if and only if both 'M' and 'E' of the day were in a scarce condition (refer to Table 4.2). Finally, WU mutation randomly swapped weekend W gene with weekly U gene in order to effect some changes for global optimal purpose.

Overall, these mutations operated over each chromosome, which in a partial random manner for a segment of a chromosome, corresponding to some constraints. In sum, the key reason for constructing directed mutation is to ensure that the constraints of a chromosome are less violated for the next generation (Ramli, 2004; Wang, Sun, Jin, Fu, Liu, Chan, & Kao, 2007).

4.5.7 Steady-state Replacement

Individuals were forced to be eliminated at this stage in order to maintain a constant population size after the evolution. This research implemented a steady-state replacement by replacing the selected parents with their offspring. Hence, the evolutionary process could at a snail's pace.

4.5.8 Stopping Criterion

Based on the discussion in Section 3.3.8, two effective stopping criteria are recommended by Ashlock (2005). They are termination based on a fixed number of generations and terminated once the expected solution achieved. The hybrid evolutionary algorithm of this research adopted the two stopping criteria. The stopping criterion of a fixed number of generations was mainly applied in evolutionary algorithm to address our scheduling and rescheduling problem. The second stopping criterion was collectively involved in the cuckoo search retrieval operator to avoid hard constraint violation. For instance, stop retrieving once the infeasible solution is solved as well as stop once a numbers of retrieved solutions are generated in the retrieval operator.

As a whole, Figure 4.17 illustrates the structure of hybrid evolutionary algorithm for nurse scheduling and rescheduling problem. Dotted rectangle highlights the newly proposed operators of the research.

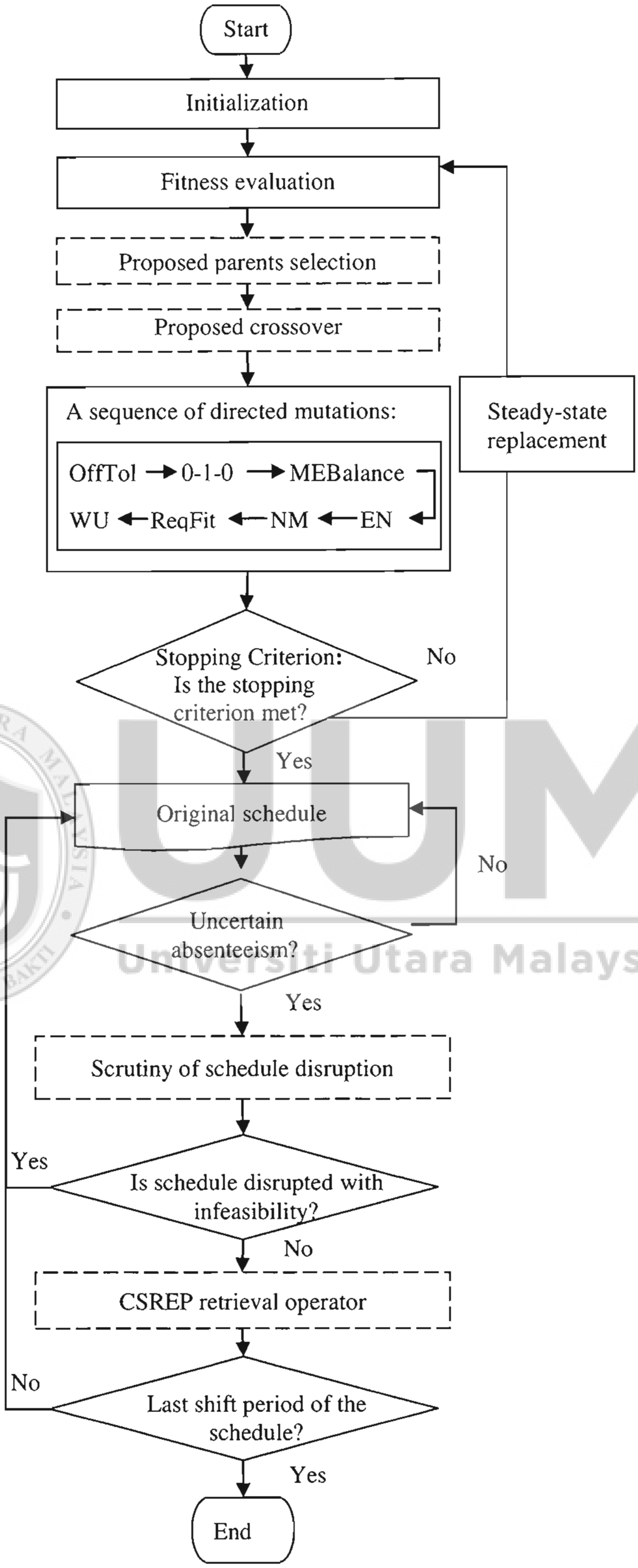


Figure 4.17. Structure chart of Hybrid Evolutionary Algorithm for nurse scheduling and rescheduling problem

Based on the structure of EA shown in Figure 4.17, there are a few ways of selecting individuals as parent and crossing over to generate offspring. The variations of the proposed models are listed in the following.

i. Modify the EA with Rank-based parent selection and 2FBlockwise crossover
ii. Modify the EA with Tournament parent selection and 2FBlockwise crossover
iii. Modify the EA with Tournament parent selection and Max[4x4]CSREP crossover
iv. Modify the EA with Maximax and Maximin parent selection and 2FBlockwise crossover
v. Modify the EA with Maximax and Maximin parent selection and Max[4x4]CSREP crossover
vi. Modify the EA with Discovery Rate parent selection and 2FBlockwise crossover
vii. Modify the EA with Discovery Rate parent selection and Max[4x4]CSREP crossover
viii. Modify the EA with Discovery Rate Tournament parent selection and 2FBlockwise crossover
ix. Modify the EA with Discovery Rate Tournament parent selection and Max[4x4]CSREP crossover

4.6 Model Evaluation

In this research, the listed nine proposed models were evaluated in terms of effectiveness, efficiency, accuracy, and reliability. Besides that, two classical models were then compared to validate the performance of all nine proposed models (refer to

Section 5.5). In sum, the performances of all total eleven models were evaluated. Originally, we tested all combinations of each parent selections and crossovers. However, some combinations seemingly performed drearily commonplace or predictably and produced infeasible solution need not be chaotic, and thus disregarded in the model comparison experiments.

A model is said to be effective when it is able to produce the best-so-far solution with the least constraint violation and lowest schedule disruption. Hence, infeasible solution is defined as a schedule has not fulfilled all the hard constraints. Deep and Thakur (2007) described three components for model evaluation: reliability, efficiency and accuracy. The basic analysis for model reliability is the percentage of success in generating a feasible solution. Model efficiency can be defined by evaluating the average number of function evaluations and average computational time, while model accuracy is justified by the mean of objective function value after a number of experiments.

Furthermore, *what-if* analysis was conducted to evaluate rescheduling. A few incidents of uncertainty created by Moz and Pato (2007) were tested. The features of the incidents tested were number of days absent, number of absent nurse, and number of consecutive days with absences. In addition, three groups of bi-weekly disruptions adapted from Moz and Pato (2007) were examined. They were *Group I* disruption during the second week, *Group II* disruption during the first week, and *Group V* disruption during the whole week. Of all, *Group V* disruption was the largest dimension of disruption with the most number of consecutive absence days. Group III and Group IV disruptions of Moz and Pato (2007) were excluded in this research because the third and fourth week disruption incidents were not suitable in our biweekly schedule.

4.7 Summary

This chapter explained thoroughly the research design used specifically in the first, second, and third phase. In addressing a nurse scheduling and rescheduling problem, a few models were developed to address the problem. In the next chapter, evaluation and validation of the models, which was the fourth phase, will be presented.

CHAPTER FIVE

EXPERIMENTS AND RESULTS

This chapter explains the ways to achieve two research objectives. These two objectives are (1) To construct an appropriate model for rescheduling purpose such that it can adjust the changes that give low impact to other nurses; and (2) To evaluate the performance of the proposed nurse scheduling and rescheduling model and the quality of the output (schedule). Toward these objectives, this chapter begins by deploying the boundary of NSRP, followed by a number comparison experiments carried out on each the proposed technique, which these results altogether were later used to find out the most fitting operators. Next, this chapter further elaborates on models comparison from among the models comprising the earlier identified fitting operators, to select the best model. Further, model validation and sensitivity analyses on disruptions are highlighted. A summary of the findings concludes this chapter.

5.1 Model Setting

As mentioned in Section 4.3, the head nurse obtained essential data on daily nurse coverage in each shift. Based on the data collected and previous studies, a lower bound number and ideal number of nurse needed for CRW and ED ward is shown in Table 5.1 below.

Table 5.1

Setting of Nurse Coverage in Daily Basis

	CRW (No. of nurse needed)				ED (No. of nurse needed)				Model 's Benchmark (No. of nurse needed)			
	Lower bound		Ideal		Lower bound		Ideal		Lower bound		Ideal	
* Total No. of Nurse	24	100%	24	100%	34	100%	34	100%	39	100%	39	100%
Morning Shift, M	4	17%	5	21%	5	15%	7	21%	6	16%	8	21%
Evening Shift, E	4	17%	5	21%	5	15%	7	21%	6	16%	8	21%
Night Shift, N	3	13%	4	17%	5	15%	7	21%	5	14%	7	19%
Senior Nurse per Shift	≥ 1				≥ 1				≥ 1			
On Call nurse (standby) per Day	1				1				1			

* Total number of nurse slightly varies in each department

Due to nurse shortage, the total number of available nurse varies throughout the year, so, we show an ideal nurse coverage rate in percentage form as well for each On Duty shift. The On Duty shifts are classified as Morning shift 'M', Evening shift 'E', and Night shift 'N'. We adjusted the coverage percentages by calculating the average percentage of nurse needed in each On Duty shift as our model's benchmark.

Next, a list of model constraints has to be obeyed for the NSRP is discussed in Section 4.3.2. The classifications of constraints involved are listed and shown in Figure 5.1.

Hard constraints:

1. 1 nurse only works 1 shift per day [*Basic rules*]
2. Nurses are required to work 6 days a week (6D1F) [*Nurse Workload*]
3. At least 16% of total nurse should be available in each M & E shift, meanwhile 14% of total nurse should be available in N shift [*Nurse coverage*]
4. At least 1 senior nurse is assigned to each work shift in a ward [*Skill classification*]
5. Work more than 6 consecutive workdays is disallowed for each nurse [*Work stretch*]
6. For each nurse, 2 off days are given to compensate for 3 consecutive Night shift duties. Hence, NNNOO shifts pattern is assigned in block according to turns and rotations [*Day off constraint*]
7. For each nurse, 1 off day is given to compensate for 1 on-call duty, if the shift of the selected nurse is originally scheduled as OFF duty shift [*Day off constraint*]
8. N-M shift forbidden in any adjacent work shifts [*Shift ordering constraint*]
9. Each compensation off day is ordered after its undertaking [*Shift ordering constraint*]
10. The reassignment of shifts must respect to the forward clockwise direction rule (M<M/E/N/Off<N/E/Off<Off/N<Off) during the implementation [*Shift ordering constraint*]
11. Nurse may not change her shift for replacement duty unless the disruption has caused the original schedule to be infeasible [*Daily adjustment rule*]
12. The delegation of on call duty shall not consider nurses who are absent with unexpected leave on that day [*Daily adjustment rule*]

Semi-Hard Constraints:

1. Total number of nurses considered as within tolerable coverage in each M & E shift should be within the range of 16% - 21%, meanwhile the total number of nurse as within tolerable coverage in N shift should be within 14%-19% range [*Nurse coverage*]
2. Split off days or single work day is discouraged for each nurse, unless the nurse has required the day off [*Work stretch*]
3. At least 1 weekend off duty is given to each nurse in a 2-week schedule. Weekend is defined as the 6th and day 7th of the week [*Weekend duty*]

Soft Constraints:

1. 21% of total number of nurses should be ideally assigned to each M & E work shift and 19% of total number of nurses is ideally needed in N shift. In this case, ideally a total of 8 nurses should be available in each M and E shift, and 7 nurses in N shift [*Nurse coverage*]
2. Each nurse should have an equal chance to execute on-call duty to avoid unfair assignment [*Nurse coverage*]
3. Equivalent total number of M and E shifts ($(M+E)/2 \pm$ (1or2)) is practiced in a personnel schedule of a nurse (by row) [*Work stretch*]
4. A roughly equivalent number of M and E shifts in a combination of M and E work stretch, with no N shifts. For 6-day work stretch, the preferred combinations of M and E are $2 \leq M \leq 4$ and $2 \leq E \leq 4$; a combination of $2 \leq M \leq 3$ and $2 \leq E \leq 3$ for 5-day work stretch; an equal balance of 2M 2E assignment for 4-day work stretch. For 2 and 3 day work stretch, it is preferred that those days have the same type of shift either 2M or 2E, and 3M or 3E [*Work stretch*]
5. The arrangement of consecutive off days shift is preferred in a personnel schedule of a nurse (by row) [*Day off constraint*]
6. Each nurse's request off application should be approved to satisfy his/her personal need [*Day off constraint*]
7. Assigning continuous shifts may respect the forward clockwise direction rule (M, E, and N in order) in scheduling [*Shift ordering constraint*]

Figure 5.1. Constraints considered

Also, a list of penalty value was set based on the importance of the considered constraints (refer to Chapter 4.3 and Figure 4.6) and the opinion of head nurses. Table 5.2 shows the detail.

Table 5.2

Penalty Value of Constraints Violation

No	Constraints violation, {*IF*}	Penalty value, {*THEN*}
1	reqFitM$_j$<16%	1000000
2	reqFitE$_j$<16%	1000000
3	reqFitN$_j$<14%	1000000
4	16%≤ reqFitM$_j$ [1] <21%	1001
5	16%≤ reqFitM$_j$ [2] <21%	1002
6	16%≤ reqFitM$_j$ [3] <21%	1005
7	16%≤ reqFitM$_j$ [4] <21%	1010
8	16%≤ reqFitM$_j$ [5] <21%	1020
9	16%≤ reqFitE$_j$ [1] <21%	1001
10	16%≤ reqFitE$_j$ [2] <21%	1002
11	16%≤ reqFitE$_j$ [3] <21%	1005
12	16%≤ reqFitE$_j$ [4] <21%	1010
13	16%≤ reqFitE$_j$ [5] <21%	1020
14	14%≤ reqFitN$_j$ [1] <19%	1001
15	14%≤ reqFitN$_j$ [2] <19%	1002
16	14%≤ reqFitN$_j$ [3] <19%	1005
17	14%≤ reqFitN$_j$ [4] <19%	1010
18	14%≤ reqFitN$_j$ [5] <19%	1020
19	SeniorNurse$_j$ (M)=0	1000000
20	SeniorNursse$_j$ (E)=0	1000000
21	SeniorNurse$_j$ (N)=0	1000000
22	SingleWorkDay$_i$ (e.g., Off-M/E/N-Off)	1000
23	MEbalance$_i$[|1|]	1
24	MEbalance$_i$ [|2|]	2
25	MEbalance$_i$ [≥|3|]	10
26	Ordering[OE]$_i$	1
27	Ordering[OM]$_i$	2
28	Ordering[MN]$_i$	1
29	Ordering[EN]$_i$	5
30	Ordering[NM]$_i$	1000000
31	WeekendOff$_i$=0	1000
32	ConsecutiveWorkdays$_i$>6	1000000
33	CellDissimilarity$_{ij}$	5
34	OnCallDelegation$_j$	10
35	ROdisapproval$_{ij}$	10
36	UntolerableRO$_{ij}$	1

There are some specific terms used in the table above. For instance, *reqFitM$_j$* signifies the requirement for 'M' shift per day. *reqFitM$_j$[1]* signifies 1 nurse away from the required 'M' shift coverage. $MEbalance_i[|1|]$ signifies 1 shift difference between total 'M' and total 'E' shifts per nurse. *CellDissimilarity* signifies the different shifts types between the original schedule and the retrieved schedule. *OnCallDelegation* signifies unfair assignment to the same nurse for on-call duties. *ROdisapproval* signifies that the nurse's request for off duty is not granted. *UnTolerableRO* signifies a number of Integrated Request Off that has not granted successfully.

5.2 Experimental Result of Population Size

The fitting of each operator is a vital part to illustrate the design of hybrid EA. If EA has taken more precise consideration in the structural exploration and exploitation, it may perform superior. The following experiments were implemented to test the fitting of the new proposed parent selections and crossovers. The population size parameter was determined for further experiments.

Table 5.3

Output of Different Population Sizes

	Population size	**10**	**12**	**14**	**16**	**18**	**20**	**30**	**40**
MM_2F	Best Fitness	4041	**2046**	4038	3046	3063	3049	4041	5043
	Average Fitness	5051	**4046**	5048	6048	5050	4050	5051	6048
	STD ('000)	0.58347	1.63479	0.53951	1.50539	1.88714	1.1877	0.69318	1.01497
	Feasible %	4/20	6/20	8/20	11/20	8/20	15/20	7/20	20/20
D_r_2F	Best Fitness	4040	3041	4044	3044	3045	3048	**2062**	4046
	Average Fitness	5051	**4047**	5045	6050	5049	4050	6050	6052
	STD ('000)	1.16397	0.91464	0.54791	1.78417	0.98651	0.99038	1.11732	1.36122
	Feasible %	3/20	8/20	6/20	12/20	12/20	15/20	13/20	16/20

Based on Table 5.3, larger population size may give higher success rate of feasibility but lack effective optimal solution. Given that, in a comparison that starting with 10 sizes difference ahead (i.e., 10, 20, 30, 40), population size of 20 was relatively the best parameter among the four groups of size. This provisional best population size of 20 obtained the same result of best average fitness 4050 and slight high feasible rate ((15/20)*100) =75% in MM_2F and Dr_2F. Also, it produced the best fitness 3049 in MM_2F but second best fitness 3048 in Dr_2F. Among the groups of 10 sizes difference, overall, a clear improvement occurred from size of 10 to size of 20 (e.g., getting lower the best fitness and average fitness) whereas inferior from size of 20 to size of 40 (e.g., getting higher the best fitness and average fitness), though Dr_2F obtained best fitness 2062 in size of 30. Due to this reason, this experiment executed more precisely for the output between population sizes of 10 to 20 (i.e., 10, 12, 14, 16, 18, and 20).

In all, this table illustrates eight groups of population size parameter each were tested by two proposed parent selection approaches. They were size of 10, 12, 14, 16, 18, 20, 30 and 40, with 100 generations. For MM2F, the population size of 12 had the lowest fitness at 2046 as well as lowest average fitness at 4046. Although the population size of 12 was the second best fitness with 3041 fitness value in D_r2F, it produced the lowest average fitness at 4047. Therefore, since we wished to search for the optimal (lowest fitness) schedule among all population sizes, the population size of 12 was determined for the following experiments.

5.3 Experimental Result of Proposed Parent Selection

There was a need to determine the most suitable relative difference between parents in discovery rate parent selection series in order to proceed with further comparisons. Figure 5.2 presents the different levels of discovery rate d_r experimented by Discovery Rate Tournament Parent Selection.

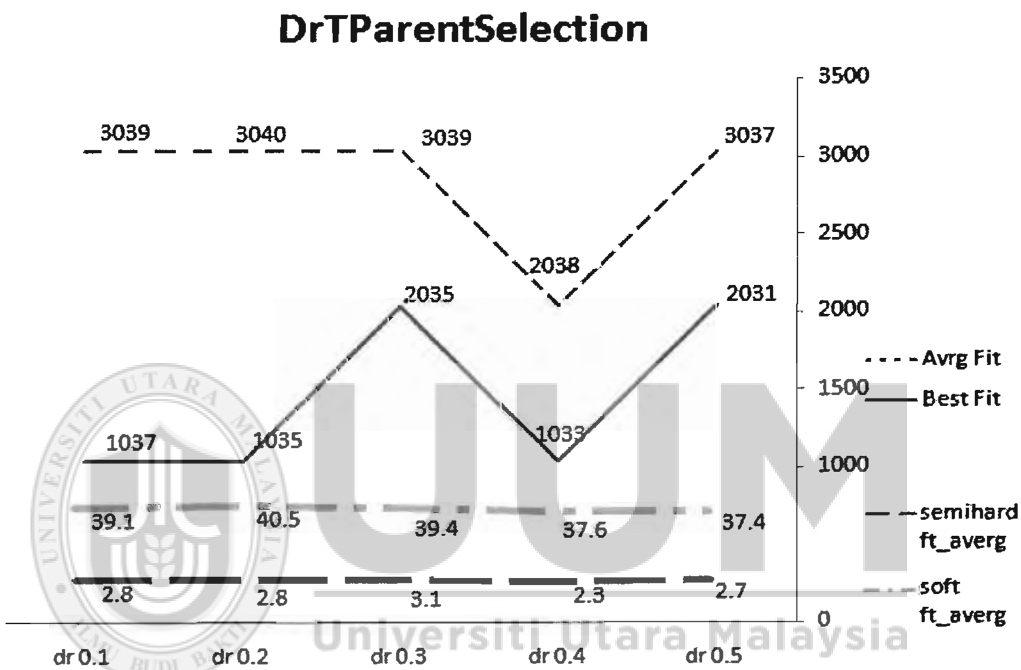


Figure 5.2. The best fitness and average fitness at different levels of discovery rate d_r in D_rT parent selection

Selective pressure may supply or preserve population diversity. Relative differnce rate is denoted as a selective pressure with the aim of determining an acceptable pressure during a selection. As Figure 5.2 reveals the fittest discovery rate d_r was 0.4, which means that the difference of the selected parents (in terms of fitness) should be at least 40%. This level of pressure found 1033 best fitness and 2038 average fitness

(2 semi-hard constraints violated and 37.6 penalty marks from soft constraints) over 20 runtimes.

From the experiment, a suitable d_r may produce better fitness and diversity for the offspring. Approximately 40% of relative difference between the two selected parents for a small population size was suggested and applied to the following experiments. In fact, d_r =0.6 had produced fast convergence. The technique may be void because it was unable to counter the excessive high selective pressure. Hence, this outcome might be caused by lower diversity in the population due to the small size of the population used. Thus in our case a high difference of selected parent at 60% was considerably negligible in this experiment.

5.3.1 Comparison of Parent Selection Operators

For a fair parent selection comparison, the experiment was implemented with the same Two-factor blockwise crossover, directed mutation, steady-state replacement strategy, and fixed parameters (e.g., 100 Generations, 12 Population size, and 30 Runs).

Table 5.4

Output of EAs with Five Different Parent Selections

Parent Selection	**MM**	**Rk**	**T**	**D_r**	D_rT
Best Fitness	2046	4062	3040	3041	**2033**
Time (Seconds)	193.55	124.55	129.77	135.23	153.07
Convergence level	97	8	15	93	39
Average Fitness	5050	6051	5049	5050	5046
STD ('000)	1.78316	1.64026	1.78316	1.06143	1.55575
Feasible rate	9/30	5/30	12/30	10/30	10/30

Overall, two parent selections (i.e. Binary Tournament T, and Rank-based Rk) and the newly proposed parent selections (i.e. Maximax and Maximin MM, Discovery Rate D_r, and Discovery Rate Tournament D_rT) were able to produce a best-so-far solution within 100 generations. Among all, D_rT parent selection achieved the best accuracy by obtaining the lowest 5046 average fitness and 2033 best fitness. Its created schedule (solution of 2033 best fitness) was able to grant all requested off days. Perhaps, the superiority of D_rT parent selection pointed toward to selecting elite parents as well as diverse characteristic of the selected parents.

As the advancement of Rk parent selection, MM parent selection was able to reduce the fast convergence of Rk parent selection and thus its feasible rate increased from 16.7% (i.e., 5/30*100) to 30% (i.e., 9/30*100). In this case, the increment signifies that the performance of offspring production depended very much on the lower selective pressure of competing number. Though, MM parent selection was weak in time efficiency (i.e. 193.55 seconds of computational time) and lower feasibility rate than T parent selection.

On the other hand, D_r parent selection produced mediocre results. Although D_r parent selection was superior to Rk parent selection in all facets, it was slightly inferior in terms of best fitness to T parent selection (i.e. 3041-3040, D_r was defeated by 1 penalty value of soft constraint violated or ((40-41)/41)= 2.4% loss of soft constraint) and the proposed MM parent selection (i.e. 3041-2046, D_r was defeated by 1 semi-hard constraint violated but ((46-41)/41)= 12.2% slightly performed better on soft constraint). Possibly, the mating strategy that merely emphasized the diversity of parental gene was not enough.

Nevertheless, one common drawback of these five different parent selection models was lack of reliability. The small feasible rates (e.g., at least 16.7% (i.e., 5/30*100) to at most 40% (i.e., 12/30*100)) indicate that they were less reliable in generating feasible solution successfully in each runs. This condition could be improved when suitable recombination operators were further taken into consideration.

5.4 Experimental Result of Proposed Crossover

Based on the population size experiment with an occurrence rate of 1.0, 0.2 pair of the total chromosomes was selected from the current population. Their corresponded components were crossed over which then went for mutation operation. In fact, this low rate was acceptable in our study. According to Montgomery and Chen (2010), in a condition if crossover rate Cr is a percentage of resultant offspring being carried to the next generation, low value of Cr may result in a search that is not just aligned with a small number of search space axes, and also behave in slow, gradual and robust. This may prevent premature convergent by gradually searching many competing optima that may then not be thoroughly explored. Thus, this could reduce our anxiety as well as Montgomery and Chen' (2010) that consistent high acceptance rate might indicate premature convergence.

Table 5.5

Setting of CSREP Crossover

Max[4x4]CSREP	**$k = 5$**	**$Tc \approx 100$**	**$Tc \approx 200$**	**$Tc \approx 300$**
R2, C2	**(2,2,5)20**	**(2,2,20)80**	**(2,2,68)272**	**(2,2,88)352**
Best Fitness	1035	2034	2036	2033
Average Fitness	3039	3040	3040	3036
STD ('000)	0.84468	0.99465	0.70103	0.51635
Feasible rate	100%	100%	100%	100%
R3, C3	**(3,3,5)45**	**(3,3,9)81**	**(3,3,30)270**	**(3,3,39)351**
Best Fitness	2033	3034	2037	1040
Average Fitness	2038	3039	3039	3042
STD ('000)	0.69930	0.70153	0.94962	1.19605
Feasible rate	100%	100%	100%	100%
R3, C4	**(3,4,5)60**	**(3,4,7)84**	**(3,4,23)276**	**(3,4,30)360**
Best Fitness	**1035**	2035	2032	1038
Average Fitness	3038	3042	4041	3042
STD ('000)	0.94969	0.94675	1.07751	1.41557
Feasible rate	100%	100%	100%	100%
R4, C3	**(4,3,5)60**	**(4,3,7)84**	**(4,3,23)276**	**(4,3,30)360**
Best Fitness	2030	2035	1036	2032
Average Fitness	3037	3038	3040	3038
STD ('000)	1.06159	1.05473	1.17646	0.87743
Feasible rate	100%	100%	100%	100%
R4, C4	**(4,4,5)80**	**(4,4,5)80**		
Best Fitness	3034	3034		
Average Fitness	3036	3036		
STD ('000)	0.00153	0.00153		
Feasible rate	50%	50%		

In this experiment, the behavioral characteristics of cells in a block which allowed REP to tackle the accessible shift patterns were on-duty shifts (e.g., morning shift and evening shift) and off-duty shifts (e.g., weekly off, weekend off, and public off). Different blocksize (*Row*, *Column*) with a total numbers of cell *Tc* (i.e., $R*C*k$) as well as crosspoints k were tested in the hybrid evolutionary algorithm. Initially, the crosspoint started at 5. Table 5.5 indicates 17 different combinations of blocks (*BlockSize(R,C), Crosspoint k)Tc* for crossing over the offspring.

Based on the output, the best fitness fell upon 5 *crosspoints* in which each blocksize was generated by vector 3*Rows* x 4*Columns*. Also, a matrix with vector 2x2 and 5 crosspoints had the same best fitness at 1035, but still little high in average fitness. Apparently, a bigger size of block was more difficult to be generated as only 50% of a feasible solution was obtained in vector 4x4. Next, by excluding the consideration of total cell *Tc*, a matrix with vector 3x4 stood out the most among all others in terms of producing the best fitness. Thus, 3x4 vector of blocksize with 5 crosspoints was applied to Max[4x4]CSREP for these reasons.

5.4.1 Comparison of Crossover Operators

For a fair crossover operators' comparison, the experiment was implemented with the same Tournament parent selection, directed mutation, steady-state replacement strategy, and fixed parameters (e.g., 100 Generations, 12 Population size, and 30 Runs).

Table 5.6

Output of EAs with Three Different Crossovers

	Rowwise	**2FBlockwise**	**Max[4x4]CSREP**
Best Fitness	3033	3040	**2033**
Time (Seconds)	114.35	129.77	210.52
Convergence level	98	15	80
Avrg Convgn rate	54	60.42	71.5
Average Fitness	4039	5049	3038
STD ('000)	1.09946	1.78316	1.28392
Feasible rate	5/30	12/30	24/30

Table 5.6 shows that 2FBlockwise crossover had greater feasibility than row-wise crossover, the one-point crossover, which was 40% (i.e., 12/30*100) over 16.7% (i.e., 5/30*100). It found that high probability of producing an acceptable solution is vital

in determining the reliability of a technique, which leads to better exploration in crossing over. This was proven by 2FBlockwise that had slower average convergence rate of 60.42 but higher feasible rate. Perhaps for that reason, 2FBlockwise had rather high 5049 average fitness.

Even so, according to the output of best fitness and average fitness, Row-wise crossover that cooperated with tournament parent selection operator performed slightly better than 2FBlockwise crossover. The difference of the best fitness was merely 7 penalty value of soft constraints violated (3040-3033=7). Thus, 2FBlockwise may be a competing model with the one-point crossover. Moreover, Max[4x4]CSREP obtained the greatest output of the best fitness (i.e. 2033) and average fitness (i.e. 3038) than row-wise crossover,the one-point crossover. Therefore, integrating cuckoo search in EA's crossover operator had improved exploitation skill by having flexible crossing points, which compromised the explored EA.

In essence, 2FBlockwise crossover operator was able to produce the approximately best fitness. The absolute blocksize of crossovers intended for matrix representation suited well in the nurse scheduling problem (NSP) but not in nurse scheduling and rescheduling problem (NSRP). For instance, in an uncertain change's condition, the need for crossing over with unshaped blocks had hindered 2FBlockwise crossover operation. Therefore, Max[4x4]CSREP was then proposed intended to fit the NSRP problem and explore a more flexible crossing over strategy.

Interruption on genes' position led to more diverse exploration and prevented premature convergence. However, some column's constraints violations were a hindrance to make more interruptions in Row-wise and 2FBlockwise. These two crossovers could be losing their flexibility and exploration search but not Max[4x4]CSREP. Table 5.6 shows 80% (i.e., 24/30*100) of the feasible rate was obtained by Max[4x4]CSREP. It strongly outperformed the one-point crossover 16.7% (i.e., 5/30*100) and even 2FBlockwise crossover 40% (i.e., 12/30*100). Also, it offered the longest search due to slower average convergence rate than that in 2FBlockwise which was 60.42 and Row-wise which was 54.

Furthermore, Max[4x4]CSREP was able to gain a fairly better result than others in terms of best fitness, better accuracy through lower average fitness and standard deviation, and highest feasibility. Yet, it was less efficient than the fastest operator of Row-wise crossover with merely 114.35 seconds needed. Max[4x4]CSREP needed approximately 210.52 seconds of computational time, which was 96.17 seconds and 80.75 seconds slower than Row-wise and 2FBlockwise, respectively. Thus, more times would be needed when executing a better exploration and exploitation search.

In sum, Max[4x]CSREP crossover was implemented by increasing the exploration and exploitation elements, 2FBlockwise was another comparable technique in terms of feasibility.

5.5 Models Comparison

The hybrid EA with three newly proposed parent selections and two newly proposed crossovers were applied to the complex nurse scheduling problem with other remaining operators staying the same. Additionally, two classical EAs (T_Row model and Rk_Row) models were implemented as the benchmark for model comparison. In all, the eleven models were:

1. T_Row : The EA with Tournament parent selection and Row-wise crossover
2. Rk_Row : The EA with Rank-based parent selection and Row-wise crossover
3. Rk_2F : The EA with Rank-based parent selection and 2FBlockwise crossover
4. T_2F : The EA with Tournament parent selection and 2FBlockwise crossover
5. T_CSREP : The EA with Tournament parent selection and Max[4x4]CSREP crossover
6. MM_2F : The EA with Maximax and Maximin parent selection and 2FBlockwise crossover
7. MM_CSREP: The EA with Maximax and Maximin parent selection and Max[4x4]CSREP crossover
8. D_r_2F : The EA with DiscoveryRate parent selection and 2FBlockwise crossover
9. D_r_CSREP : The EA with DiscoveryRate parent selection and Max[4x4]CSREP crossover
10. D_rT_2F : The EA with Discovery Rate Tournament parent selection and 2FBlockwise crossover
11. D_rT_CSREP : The EA with Discovery Rate Tournament parent selection and Max[4x4]CSREP crossover

Table 5.7

Output of All Eleven EA Models' Comparison

	T_ Row	**Rk_ Row**	**Rk_ 2F**	**T_ 2F**	**T_ CSREP**	**MM_ 2F**	**MM_ CSREP**	**D_r_ 2F**	**D_r_ CSREP**	**D_rT_ 2F**	**D_rT_ CSREP**
Best Fitness	3033	4040	4062	3040	2033	2046	1041	3041	2032	2033	**1033**
UnTolerableRO	1	0	1	1	1	4	2	3	0	1	0
NoDisapprovalRO	0	0	1	0	0	0	0	0	0	0	0
Time (Seconds)	114.35	189.2	124.55	129.77	210.52	193.55	168.56	135.23	192.19	153.07	169.03
Convergence level	98	9	8	15	80	97	83	93	72	39	78
Average Fitness	4039	4040	6051	5049	3038	5050	3037	5050	3039	5046	2038
STD ('000)	1.09946	-	1.64026	1.78316	1.28392	1.78316	1.13679	1.06143	0.7479	1.55575	1.06696
Feasible rate	5/30	1/30	5/30	12/30	24/30	9/30	25/30	10/30	30/30	10/30	30/30

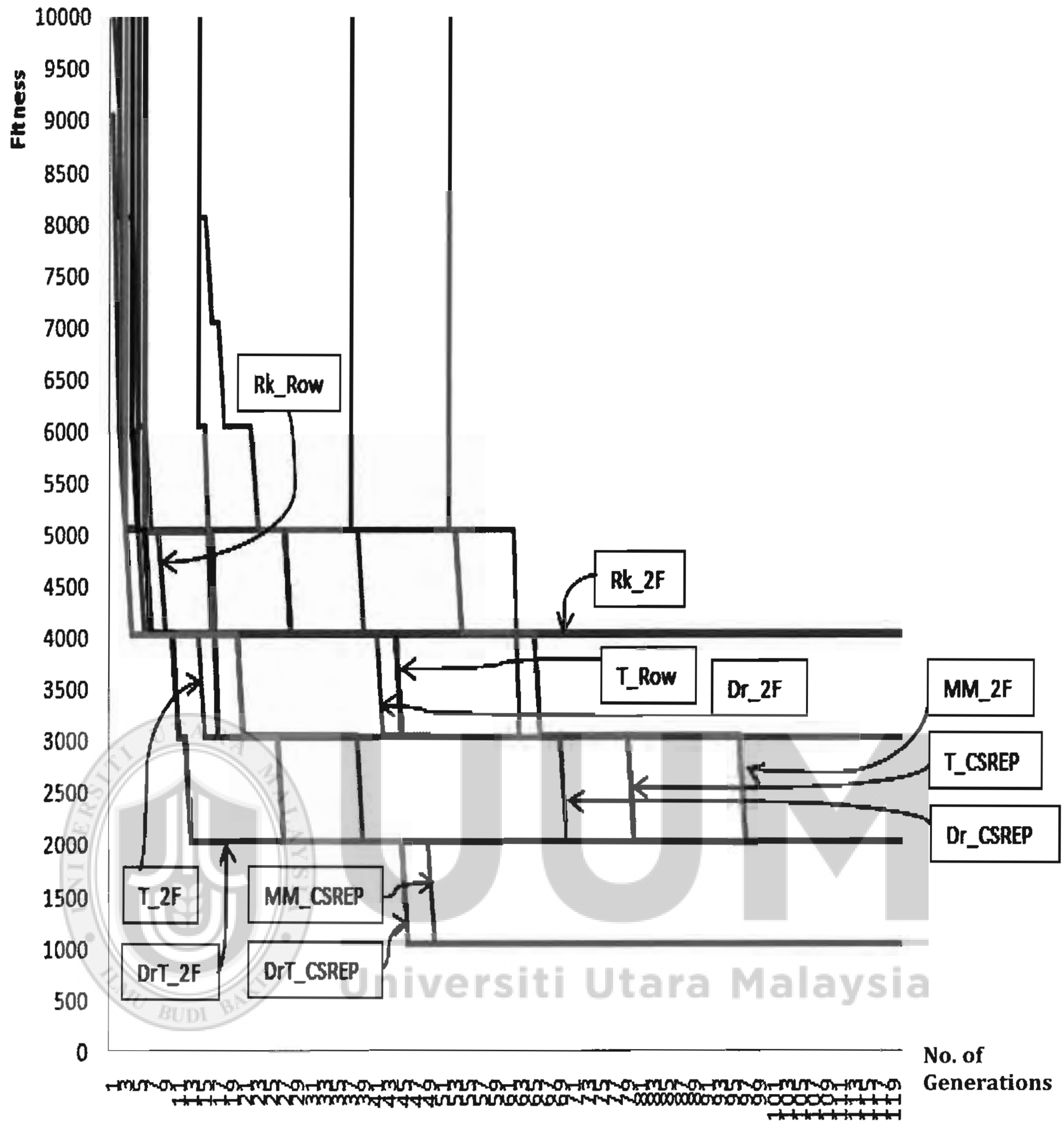


Figure 5.3. A convergence graph of all eleven EA models for comparison

Based on previous results, D_rT was found to be the finest parent selection operator and Max[4x]CSREP the finest crossover. For model effectiveness, D_rT_CSREP's best fitness 1033 was the lowest value among all models, as shown in Table 5.7. This fitness value indicated that none of the hard constraint was violated; however, at 1 semi-hard constraint was violated and 33 penalty value was given for soft constraint. D_rT_CSREP was the most accurate model with the lowest 2038 average fitness and a standard deviation 1.06696. MM_CSREP was found to rank the second, given its best fitness 1041. However, MM_CSREP obtained higher average fitness 3037, higher standard deviation 1.13679, and lower feasible rate, which reduced its accuracy and reliability. Additionally, with 95% of confidence interval, there was a significant difference in average fitness (penalty value) across the different EA hybrid models, $F(10, 150) = 9.518$, *p-values* of 0.000. Tukey HSD test was used to further determine the difference. It was found that the best model D_rT_CSREP was significantly different from other models (i.e. Rk_2F, T_2F, MM_2F, D_r_2F, and D_rT_2F). The homogeneity of variance assumption with *p-value* 0.04 ($P > 0.05$) was little violated. In fact, the ANOVA test was still statistically relevant due to only malt variation from the assumption. Hence, any interpretation of the main effect must be undertaken with caution.

To confirm the models' reliability, Rk_Row was less likely to produce an acceptable solution since it had the lowest feasible rate of 3.33%, followed by T_Row and Rk_2F (16.67%). It concluded that these two classical models, especially Rk-Row, were weak in exploration search thus hard to support the recent complex NSP. At the other end of the continuum, D_r_CSREP and D_rT_CSREP obtained 100% feasible rate. This was even superior to T_Row and Rk_Row models. Even so, the 2F-based models such as T_2F, MM_2F, D_r_2F, and D_rT_2F attained a fairly better feasible rate than the two classical EAs. Besides Rk_2F, the other models' feasible rates were in between 30% and 40%. In sum, it can be concluded that CSREP is a promising model in terms of reliability capable for producing a feasible solution. In other words, the hybrid EA worked effectively.

Based on the performance of model T_2F and model Rk_2F in terms of the best fitness and average fitness, these classical parent selection and 2FBlockwise comprised models were inferior to the benchmark model T_Row. Table 5.7 shows the best fitness of T_2F and Rk_2F are 3040 and 4062 which higher than T_Row's 3033. There was also similar condition to the average fitness where T_2F's 5049 and Rk_2F's 6051 were higher than T_Row's 4039. However, there is also room for improvement to the 2FBlockwise-based models since they obtained greater feasibility rate (e.g., (12/30*100)=40% > (5/30*100)=16.7%). Therefore, the 2FBlockwise-based models could be improved by replacing classical parent selection with new proposed parent selections. This was due to new proposed selections experienced with dissimilarity relationship. The new 2FBlockwise-based models MM_2F, D_rT_2F, and D_r_2F produced high average fitness 5050, 5046, and 5050 with big value of standard deviation 1783.16, 1555.75 and 1061.43, respectively. Though big value of standard deviation was said to have unstable performance, yet, the instability or the big value was also alluding to the dispersion of all feasible fitness values from the average fitness. Therefore, in the big gap of standard deviation, these three 2F-based models eventually produced lower best fitness value (i.e., 2046, 2033, and 3041), which respectively better than or comparable with T_Row's 3033 best fitness. As a conclusion, MM_2F and D_rT_2F were superior to T_Row, Rk_2F, and T_2F. The slight underperformance of D_r_2F was because of the essence of D_r parent selection and 2FBlockwise crossover, both of which skewed more to exploration principle. Thus, lack of exploitation in this model made hard to identify local optimal.

In minimization problem, the complex NSP which required higher nurse coverage with timely preferences was successfully achieved, except Rk_2F. All requested off days were approved, as indicated by the value '0' in *NoDisapprovalRO*. Also, each model was able to produce at least one acceptable schedule that was free from hard constraint violations, at most 4 semi-hard constraint violations, and less than 62 penalty value of soft constraint violations. Of all, the optimal solution of Rk_Row, D_r_CSREP, and D_rT_CSREP had attained the Integrated Request Off concept completely (i.e., *unTolerableRO*= 0). They were simultaneously fulfilled both head nurse and nurses' desires on timely off duty. It is ideal if the solution had a lower value of best fitness with zero value of *unTolerableRO*. Thus, D_rT_CSREP stood out among the three.

With regards to computation time, T_Row was the fastest with approximately 2mins and the slowest model was T_CSREP with approximately 3.5mins. The difference in the execution time was just about 1 minute 36 seconds. However, time evaluation for NSP was not as important as in NRSP because NRSP involved unplanned aspects.

Based on Figure 5.3 that shows the best solution of each model, Rk_2F and Rk_Row had premature convergence that was stuck during early generation runs. A little slower convergence but better solution than Rk_2F and Rk_Row were D_rT_2F and T_2F. Despite having fast convergence, they obtained a weak solution. T_Row, MM_2F and D_r_2F had slow convergence at around generation 97 and produced fairly weak solutions. On the other hand, D_rT_CSREP which obtained the lowest fitness solution had converged at generation 78. MM_CSREP, D_r_CSREP, and

T_CSREP which had a similar range of convergent rate with D_rT_CSREP produced mediocre results. They obtained fairly low average fitness, as show in Table 5.7. In our case, the model that converged at the middle of the generations had a balance of exploration and exploitation.

As a conclusion, high selective pressure towards elite parents may probably result in premature convergence and hence losing population diversity since EA with Rank-based parent selection had a great impact on fast convergence, as indicated at 8th generation runs. In that sense, more exploration produced better performance. To note, D_rT_CSREP stood out by obtaining the best feasible rate, least average fitness and excellent in timely preferences.

5.6 Retrieval Validation

Uncertain absenteeism occurs at work. Thus, sensitive analysis was conducted with CSREP retrieval operator for the following rescheduling validation. As mentioned in Section 4.6, several biweekly disruption instances adapted from Moz and Pato (2007) has are in Table 5.8. Note that the II.1_32 and V.3_32 instances of Moz and Pato (2007) were excluded in this test because they had similar instances as II.2_32 and V.1_32.

Table 5.8

Features of the Disruption Instances

Instances	The first day of absence	No. of absent nurses	No. of consecutive days with absences
I.1_32	13	2	2
I.2_32	12	2	3
I.3_32	10	3	5
I.4_32	8	1	1
I.5_32	10	4	5
I.6_32	11	9	4
I.7_32	12	10	3
II.2_32	6	1	1
II.3_32	5	1	3
II.4_32	2	1	8
II.5_32	3	8	12
II.6_32	6	6	9
II.7_32	4	4	8
II.8_32	2	3	8
V.1_32	1	9	14
V.2_32	1	10	14
V.4_32	1	15	14

For the NSRP, this research focused upon decision making during disruptions, quality change of schedule adjustment and reducing quantity change for retrieval. Table 5.9 shows the output of all the disruption instances. Several evaluation items for a retrieved schedule are below:

NonRetrieval: Rescheduling due to uncertain absenteeism in a nurse's schedule through pre-retrieval. Generally, it is suitable for some small disruption cases.

Retrieval: Rescheduling due to uncertain absenteeism that involves the absented nurse and other nurses, by using retrieval operator. Generally it is suitable for serious disruption cases.

RO resolver:	Estimated successful rate that employs requested nurses on off days to fill in for insufficient coverage during disruption
xRTolerc:	Number of unsuccessful Integrated Request Off during disruption that continues from *unTolerableRO* during scheduling phase.
Rdisappv:	Number of disapproval of requested off duties in rescheduling during disruption that continues from *noDisapprovalRO*in scheduling phase.
TotChgCell:	Total number of cells changed
XFairDelg:	Total number of unfair on-call delegation
Time (Seconds):	Computation time

Table 5.9

Output of All the Disruption Instances

GROUP I		I1_32	I2_32	I3_32	I4_32	I5_32	I6_32	I7_32
Scheduling	Best Fitness	1033	1037	3038	3035	2034	2041	1033
	unTolerableRO	1	2	2	3	2	3	5
	noDisapprovalRO	0	0	0	0	0	0	0
Rescheduling	Retrieval (%)	43%	20%	50%	3%	60%	100%	93%
	Best Fitness	3039	3043	4047	4045	5041	4078	5042
	xRTolerc	1	2	2	3	2	3	5
	Rdisappv	0	0	0	0	0	0	0
	TotChgCell	3	3	4	2	5	14	11
	XFairDelg	0	0	0	0	0	0	0
	Time (Seconds)	0.916763	2.581827	0.873508	1.913535	0.906276	4.255609	3.412228
	AvrgRFitChg	2011	1011	3012	1010	4015	7047	5057
	STD(RfitChg)	799.27	891.98	1099.90921	_	1350.85	928.05	1351.345
	oriSchedule	2032	1034	3038	1036	2036	-	2035
	Best nonRetrieval Fitness	3031	2035	3049	2036	3036	-	7038
	Time (Seconds)	2.148416	0.180973	0.486191	0.177628	0.523539	-	0.28395
	Feasible rate (%)	100%	100%	100%	100%	100%	100%	100%
	RO resolver success rate (%)	100%	100%	100%	100%	100%	70%	87%
GROUP II		**II2_32**	**II3_32**	**II4_32**	**II5_32**	**II6_32**	**II7_32**	**II8_32**
Scheduling	Best Fitness	5041	3040	2039	4037	1037	3031	2034
	unTolerableRO	2	4	3	3	1	2	2
	noDisapprovalRO	0	0	0	0	0	0	0
Rescheduling	Retrieval (%)	7%	13%	13%	100%	100%	60%	43%
	Best Fitness	5046	3045	2044	13140	6104	5047	3045
	xRTolerc	2	4	3	3	1	2	2
	Rdisappv	0	0	0	0	0	0	0
	TotChgCell	2	2	2	19	13	5	5
	XFairDelg	0	0	0	0	0	0	0
	Time (Seconds)	1.118648	2.333321	0.996036	2.13493	5.019985	1.086927	0.942412

Table 5.9 continued

	AvrgRFitChg	5	1007	1008	14091	7052	6020	3012
	STD(RfitChg)	0	959.26	820.992489	2469.54	1834.529	2734.856	1963.902
	oriSchedule	1039	1033	1033	-	-	3038	2039
	Best nonRetrieval Fitness	1041	1031	2034	-	-	7041	4042
	Time (Seconds)	0.549632	0.198307	2.122604	-	-	0.175382	0.212957
	Feasible rate (%)	100%	100%	100%	100%	100%	100%	100%
	RO resolver success rate (%)	93%	100%	100%	97%	97%	97%	100%

GROUP V		**V1_32**	**V2_32**	**V4_32**
Scheduling	Best Fitness	2039	2035	1032
	untTolerableRO	1	2	1
	noDisapprovalRO	0	0	0
Rescheduling	Retrieval (%)	100%	100%	100%
	Best Fitness	12116	15115	25537
	xRTolerc	1	2	2
	Rdisappv	0	0	0
	TotChgCell	21	20	94
	XFairDelg	0	0	2
	Time (Seconds)	5.18206	1.758692	20.161886
	AvrgRFitChg	14073	17147	28447
	STD(RfitChg)	2409.17	1693.9867	2751.38023
	oriSchedule	-	-	-
	Best nonRetrieval Fitness	-	-	-
	Time (Seconds)	-	-	-
	Feasible rate (%)	100%	100%	100%
	RO resolver success rate (%)	73%	87%	100%

5.6.1 Disruption Impact on Preferences

Output in Table 5.9 was produced after 30 run times to show the best solution of each group of disruption instances, named as Group I, Group II and Group V. What-if analysis of the disruptions was required to observe the schedule performance in terms of high nurse preferences and fairness in on-call delegation during uncertainty. Overall, the retrieval operator had performed well in Group I disruption, Group II disruption, and Group V disruption. Based on the output, all requested off duties were fully granted as well as in a manner of not degenerated the Integrated Request Off which was highly preferred. Rescheduling could remain the Integrated Request Off as similar as planned in scheduling phase, except case V4_32. Only one Integrated Request Off failed with separation (i.e. *xRTolerc* – *unTolerableRO* =2-1 =1). However, the *Rdisappv* for all cases in the three groups were equal to zero and the number of *xRTolerc* was similar to that of *unTolerableRO*. In the combination of scheduling and rescheduling, this can be concluded that the schedule stability of the nurse preferences was achieved even during rescheduling.

Furthermore, fairness in nurse delegation for on-call duty almost achieved a zero defect in the three groups of disruptions, except case V4_32 of Group V. The value of 2 in *XFairDelg* indicated that two nurses were assigned to on-call duties more than the average. However, the retrieval operator was capable to give fair delegation to most of the disruption cases since the overall outputs of *XFairDelg* in all other cases were equal to zero.

Based on the above assessment, V4_32 was a bit challenging to the retrieval operator because of several nurses concurrently took long consecutive absences from work.

The result shows that the head nurse needs to be aware of two high preferences which are fair on-call delegation and Integrated Request Off during a larger dimension of uncertain disruption. Although this case V4_32 had went through the most severe retrieval process among all cases which obtained the highest penalty value of best retrieved fitness 25537 when its original schedule's best fitness was 1032, the feasibility which is a vital concern in rescheduling was achieved.

Besides investigating the impact on preferences, the contingency adjustments of rescheduling (i.e., pre-retrieval and retrieval) also considered to the seriousness level of an uncertain disruption. It was studied next.

5.6.2 The Seriousness of Disruption

The seriousness of various disruptions adopted from Moz and Pato (2007) on the original schedules can be observed by the retrieval percentage and fitness changes, as shown in Table 5.9, Figure 5.4, Figure 5.5, and Figure 5.6. These figures show before and after rescheduling under various disruption incidences (i.e. Group I, II, and V). Fitness change, an output of rescheduling, explains the adjustment (i.e., the difference between the original schedule and the retrieved schedule) during uncertainty. The graph of fitness change clearly shows the impact of disruption to an original schedule in terms of semi-hard and soft constraint violations, in which the adjustment was dealt as at the pre-retrieval process or retrieval operator process. In the graph, the runs before an arrow '⬆' is the rescheduling that dealt by retrieval operator. On the flipside, starting from the arrow to the end of the runs is the rescheduling that dealt as at pre-retrieval process.

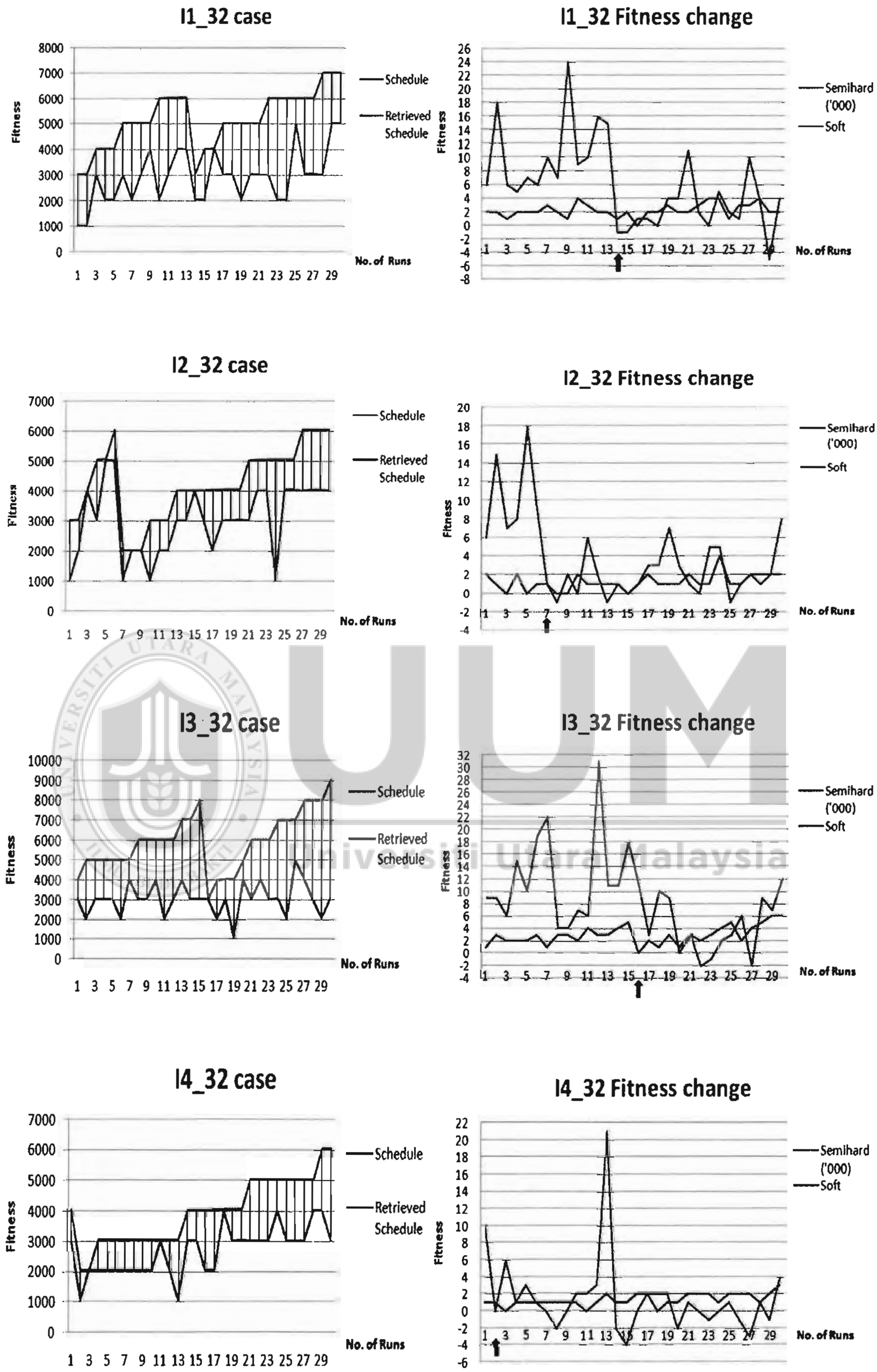


Figure 5.4. Retrieval cases of Group I disruptions and their fitness changes

Figure 5.4 continued

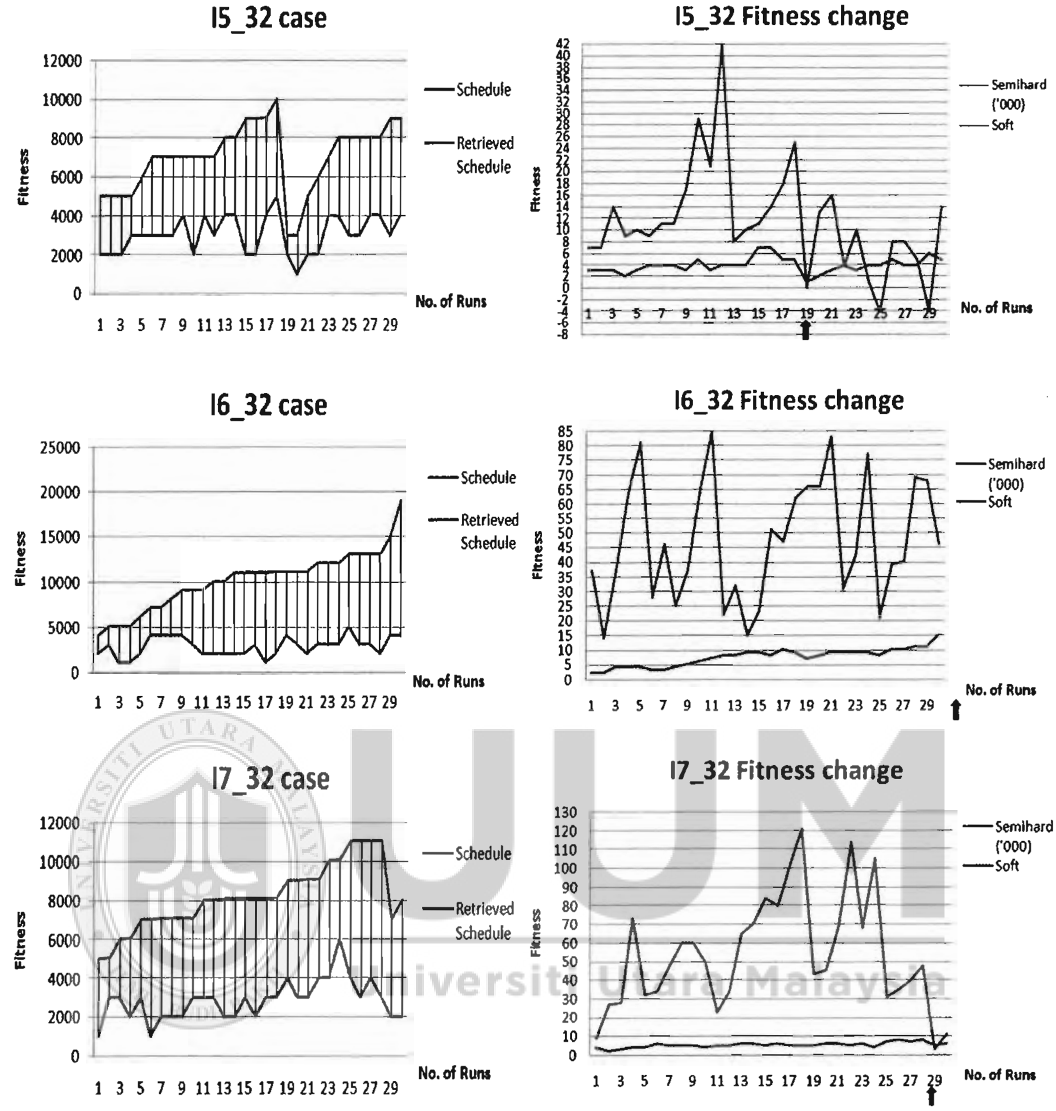


Figure 5.4 illustrates the overall rate of retrieval and fitness changes in Group I disruption (i.e.I1_32, I2_32, I3_32, I4_32, I5_32, I6_32, and I7_32). Group I disruptions is a disruption occurred at the end week of a schedule. This limits the rescheduling within a smaller search space. In these disruptions, our rescheduling model determined the seriousness of a schedule disruption through pre-retrieval (i.e., process before retrieval) in order not to retrieve the schedule impulsively, and thus preserved the original schedule. It was an effective contingency move. As shown in the seven graphs of fitness changes in Figure 5.4, the output of rescheduling that

dealt as at pre-retrieval (i.e., after an arrow) were mostly better (i.e., lesser constraint violations) than the output that dealt by $CSREP_r$ retrieval operator (i.e., before the arrow). Particularly, cases I1_32, I3_32, and I5_32 depicted the result clearly. However, this does not mean that $CSREP_r$ was underperforming; it reflects that the schedule was ready for any serious disruptions, given that some light disruptions could be solved as at pre-retrieval.

I4_32 case was considered the lightest disruption among the cases of Group I. It took only 1/30 run (i.e. 3% retrieval rate) that used retrieval operator for rescheduling. The second lightest disruption case was I2_32 that took 6/30 runs (i.e. 20% retrieval rate) by $CSREP_r$ retrieval operator. In these cases of light disruption, there were a few fitness changes in semi-hard and soft constraint violation. The output shows low average fitness change of 1010 and 1011 when tackling I4_32 case and I2_32 case, respectively (see Table 5.9).

On the other hand, I6_32 case was the worst disruption in Group I as it obtained 100% retrieval rate (i.e. 30/30 runs) and the highest average fitness change 7047 (see Figure 5.4 and Table 5.9). In this case, approximately 2 to 15 semi-hard constraints and 14 penalty values to 85 penalty values of soft constraints were violated. Subsequently, the disruption of I7_32 case had almost the same seriousness as I6_32 case which obtained 93% retrieval rate. In these cases of serious disruption, there was a bigger fluctuation of soft constraints violation but quite a constant semi-hard constraints violation. However, the retrieval did not ignore nurse timely preferences (i.e. $Rdisappv = 0$) and fair assignment of on-call duty (i.e. $XFairDelg = 0$), as shown in Table 5.9.

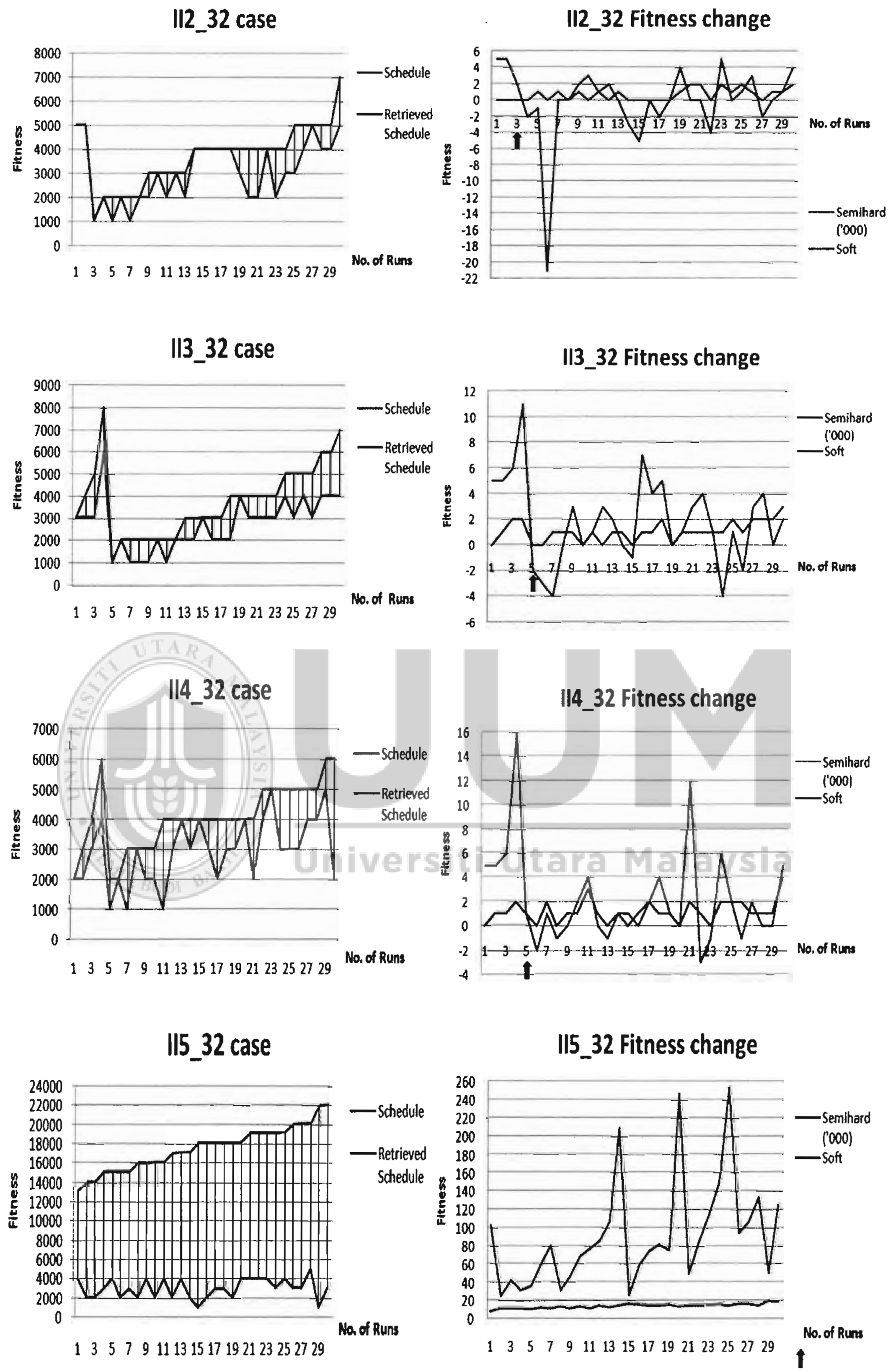


Figure 5.5. Retrieval cases of Group II disruptions and their fitness changes

Figure 5.5 continued

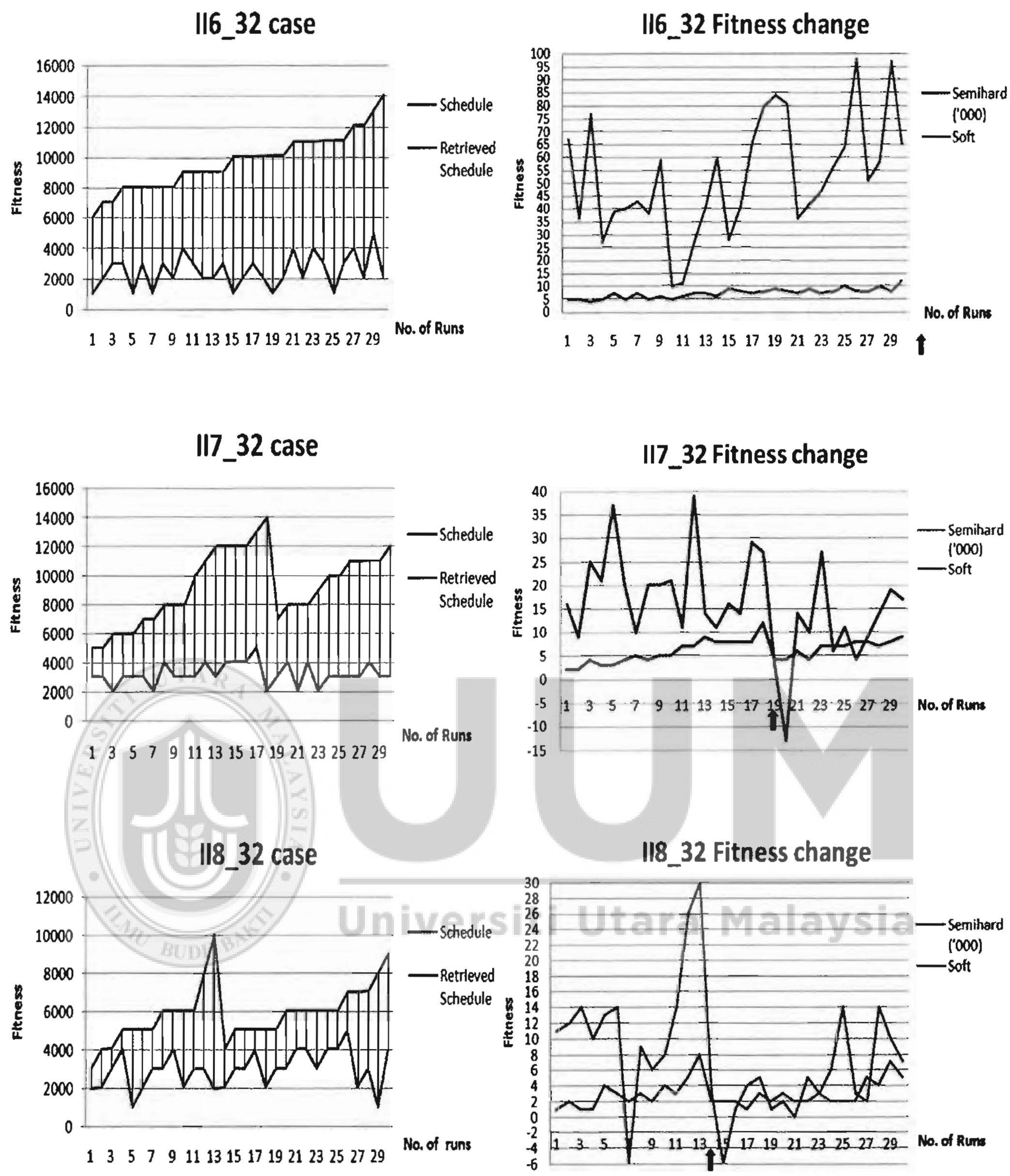


Figure 5.5 illustrates Group II disruptions that occurred in the first week of the schedule in terms of retrieval and fitness changes (i.e.II2_32, II3_32, II4_32, II5_32, II6_32, II7_32, and II8_32). In this group, II2_32 case, II3_32 case, and II4_32 case were considered as light disruptions which were mostly solved as at pre-retrieval with lower fitness changes. Cases as II5_32 and II6_32 had the most critical disruptions in Group II in which both were retrieved in all runs with high average

fitness changes 14091 and 7052 respectively (see Figure 5.5 and Table 5.9). The fitness change graph shows that II5_32, which was the worst case among Group I and II, showed approximately 10 to 20 of semi-hard constraints and 30 to 250 penalty value of soft constraints were violated. Yet, the feasibility of II5_32 case was not disappointing; II5_32 case in Figure 5.5 depicted a minimum fitness value of retrieved schedule at 13140 and its maximum fitness value at 22165 with the overall feasible rates of 100% (i.e. 30/30 runs).

Note that Group I disruption was more critical than Group II's in terms of the overall retrieval rate. This might be due to the total number of absent nurses in all cases. In Group I, there were 31 absent nurses, which was higher than Group II (i.e. 24 absent nurses), as shown in Table 5.8. Even though Group II had greater disruption effect than Group I in terms of average fitness change (see outputs of *AvrgRFitChg* in Table 5.9), when comparing I5_32 case and II7_32 case, both had the same number of absent nurse disruption (i.e. 4 absent nurses). In addition, I5_32 case had a higher retrieved rate than II7_32 despite the fact that the overall disruption of I5_32 case was less severe than II7_32 case. Apparently, retrieval was less needed in Group II when compared to Group I that had smaller search space.

As a conclusion, for Groups I and II, by observing the fitness changes before and after the arrow, the fitness changes of pre-retrieval (after arrow) were mostly lower than the retrieval change (before arrow), as evidenced in I1_32, I2_32, I3_32, I4_32, I5_32,II4_32,II7_32 and II8_32. Additionally, even some soft constraint violations were reduced after rescheduling and these can be noticed at pre-retrieval in I1_32, I2_32, I3_32, I4_32, I5_32, II2_32, II3_32, II4_32, II7_32 andII8_32. The result revealed the advantages of having pre-retrieval to scrutinize the seriousness of the disruption before retrieval was conducted. In this way, the schedule stability could be maintained by involving pre-retrieval. Furthermore, retrieval operator was meant to overcome more serious disruption.

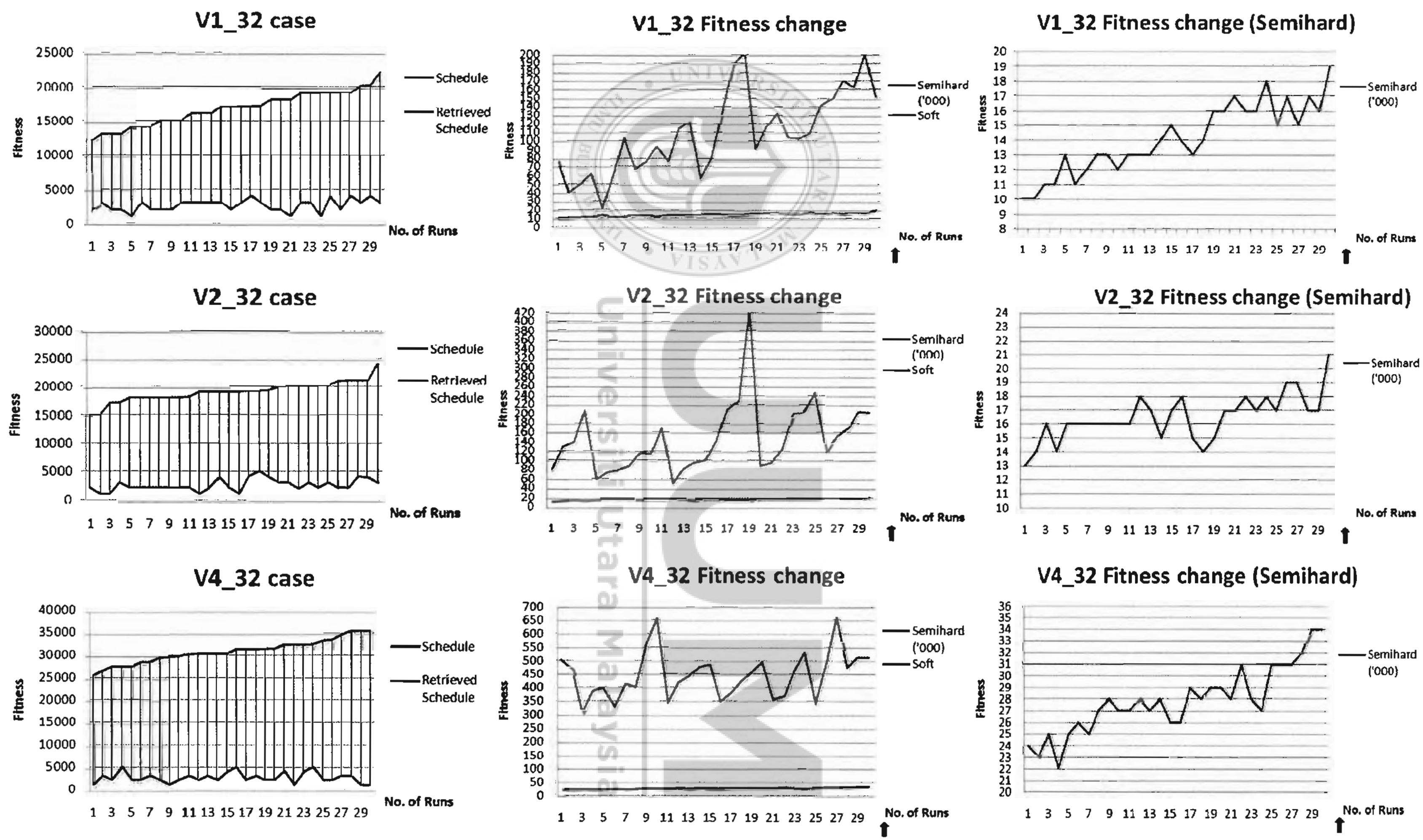


Figure 5.6. Retrieval cases of Group V disruptions and their fitness changes

Figure 5.6 illustrates Group V disruptions (i.e.V1_32, V2_32, and V4_32) in terms of retrieval and fitness changes in view of soft and semi-hard constraints violation. The fitness changes in semi-hard constraint violation were shown in the last column. Group V had a larger dimension of disruption. These worst disruption conditions can be real as several nurses will be on maternity leave. The long absent days can have a great impact on the schedule. This is reflected by the approximately 100% retrieval rate for the three disruption instances.

Figure 5.6 shows that the fitness of retrieved schedule had a positive relationship with the number of absent nurse. The best fitness of retrieved schedule for V1_32, V2_32, and V4_32 cases were 12116, 15115, and 25537 when tested with 9, 10 and 15 absent nurses, respectively (see Table 5.8). In all runs of the three cases, the fitness of the retrieved schedule was constantly higher than the original schedule. The graphs of the fitness change show that the number of semi-hard constraint violations in V1_32, V2_32, and V4_32 were approximately 10 to 19, 13 to 21, and 22 to 34, respectively, whereas soft constraint violations arose with 20 to 200, 50 to 420, and 300 to 650 penalty values.

Even though schedule feasibility is vital in rescheduling for a ward operation, the result showed that retrieval causes higher fitness. Therefore, the criticality of searching optimal solution in rescheduling is not less than those in scheduling.

As a conclusion to the seriousness of disruption in Group I, II and V, the fitness change in retrieval and pre-retrieval indicated that not all disruptions shall be retrievable. Thus, unstable schedule can be improved by making the schedule ready for disruption. As rescheduling involves retrieval, this will produce few changes to the schedule. Furthermore, rescheduling considers real-time conditions in the original schedule which involves past, current, and future shifts. In all, these show that the combination of scheduling and rescheduling is needed. In addition, Group I disruption in the second week was slightly difficult to be solved than Group II disruption in the first week. This means that retrieving process within a smaller search space is more difficult. The challenge, therefore, is producing quality schedule adjustment and not simply the number of changed cell in retrieval.

5.6.3 The Quantity and Quality Changes of Disruption

As mentioned in Section 4.3.2.9, Table 5.10 shows the quantity and quality change to the original schedule during disruption. It shows the computational output of Moz and Pato (2007) and our retrieval operator, as well as the result between non-radical change and radical change.

Table 5.10

Computational Result of Moz and Pato (2007) Model and CSREP$_r$ Model

Instances	Moz and Pato (2007)			CSREP$_r$				
	Best (Cell)	Worst (Cell)	Time (Seconds)	Best (Cell)	Worst (Cell)	Time (Seconds)	Non-radical change of retrieval	Radical change of retrieval
I1_32	5	**5**	186.77	**3**	7	0.916763	**11/13**	7/13
I2_32	5	**5**	229.67	**3**	9	2.581827	**5/6**	2/6
I3_32	8	**8**	360.87	**4**	11	0.873508	**12/15**	6/15
I4_32	3	3	508.67	**2**	**2**	1.913535	1/1	1/1
I5_32	10	**10**	367.44	**5**	14	0.906276	**17/18**	5/18
I6_32	15	**15**	326.51	**14**	26	4.255609	7/30	**12/30**
I7_32	**10**	**10**	244.74	11	25	3.412228	7/28	**14/28**
II2_32	5	**5**	635.1	**2**	6	1.118648	**2/2**	1/2
II3_32	7	7	710.68	**2**	**3**	2.333321	3/4	3/4
II4_32	13	**13**	939.87	**2**	17	0.996036	**4/4**	3/4
II5_32	20	**20**	894.63	**19**	52	2.13493	**10/30**	7/30
II6_32	24	**24**	724.12	**13**	44	5.019985	**22/30**	5/30
II7_32	10	**10**	778.75	**5**	33	1.086927	**16/18**	6/18
II8_32	11	**11**	929.45	**5**	41	0.942412	**11/13**	5/13
V1_32	**14**	**14**	2167.46	21	78	5.18206	**10/30**	8/30
V2_32	27	**27**	2587.85	**20**	80	1.758692	**9/30**	7/30
V4_32	144	166	4616.67	**94**	**114**	20.16189	**11/30**	6/30

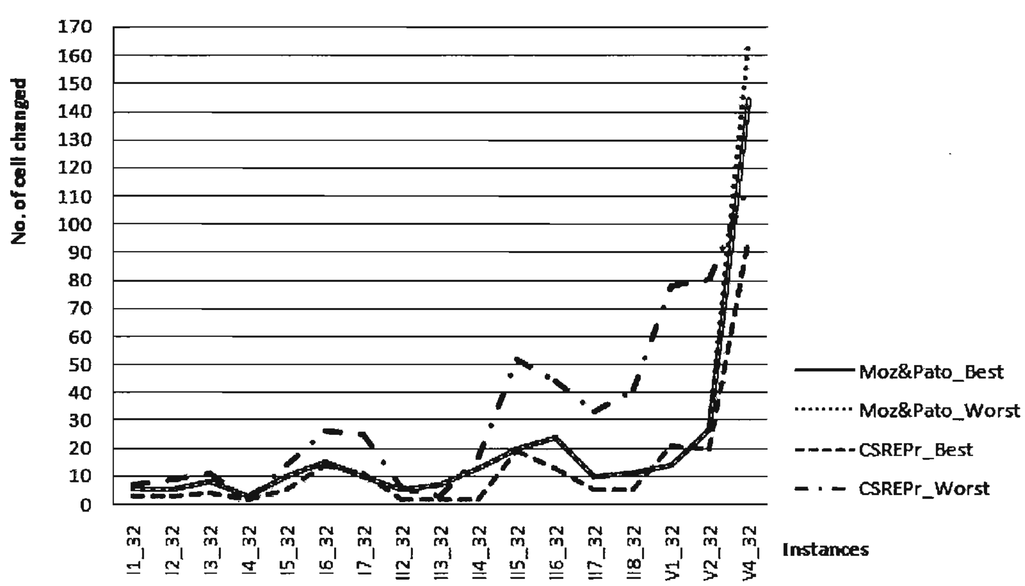


Figure 5.7. Quantity change of disruption instances

Based on the best solution of total number of changed cell, $CSREP_r$ retrieval operator performed better than Moz and Pato (2007)'s genetic algorithm hybridization. Table 5.10 shows that $CSREP_r$ had lesser changed cells in most of the disruption instances, though there were slightly inferior to I7_32 and V1_32. In some light disruption incidents such as I1_32 to I5_32 and II2_32 to II4_32, the worst solution of $CSREP_r$ was approximately close to the worst solution of Moz and Pato's (2007) (see Figure 5.7). In the heaviest disruption, the best retrieval of V4_32 involved 94 cells changed and the worst retrieval 114 cells changed. Though little change to the original schedule was achieved, it would be meaningless if the original schedule was not planned well.

In fact, this experiment might less compare squarely since Moz and Pato (2007) had a different set of assumptions in a rescheduling problem. The key difference was the different original schedules that had uncertain reaction to the disruption instances, as shown in Section 5.4.2. Moz and Pato formulated the rescheduling problem in a multi-level directed network. Therefore the solution was counted by the number of arcs swap between a set of disjoint paths. However, since both models focused on the quantity change which essentially calculated the cell dissimilarity between the original schedule and the retrieved schedule, we could make a comparison between them. Moreover, the feasibility problem of Moz and Pato (2007) was constrained to no day off is assigned in a sequence of seven days. As such, both models were capable in producing a feasible solution for the disruption incidents. However, $CSREP_r$ was superior in terms of computation time, clocking less than 20 seconds for each disruption instance, as shown in Table 5.10. Hence, $CSREP_r$ can be said to be an efficient retrieval operator. This is a significant consideration since computation time is another important element in rescheduling.

In order to consider both quality and quantity change of retrieval, the output of the non-radical change and radical change in Table 5.10 were gained by concurrently obtained the lowest fitness value and lowest total number of changed cell. In all, non-radical change of retrieval was outperformed in many types of disruption. Note that radical change method stood out in I6_32 and I7_32 with 40% (i.e. 12/30*100) and 50% (i.e. 14/28*100), respectively. This suggests that radical change of retrieval may be a good resolution when experiencing a heavy disruption on a smaller search space condition.

5.7 Summary

In this research, nurse scheduling and nurse rescheduling were integrated by hybrid evolutionary algorithm. The former was strategic while the latter was operational. Output of scheduling was the input for rescheduling, and vice verse. In other words, the output of each stage had mutually generated and applied as each stage's input data. Specifically, objective two and five were then completed by evaluating and validating a few models. The models were T_Row, Rk_Row, Rk_2F, T_2F, T_CSREP, MM_2F, MM_CSREP, D_r_2F, D_r_CSREP, D_rT_2F, and D_rT_CSREP.

Of all, D_rT parent selection and CSREP crossover were the best among the operators. Also, D_rT parent selection complemented well CSREP crossover whereby only 1033 best fitness solution was obtained in scheduling. This means that a balance of exploitation and exploration can improve search. Extreme exploitation can lead to premature convergence and intense exploration can make the search ineffective.

In addition to the data analysis, discussions about rescheduling response towards disruption instances were presented in this chapter. The vital concern of rescheduling was achieved in the pre-retrieval and retrieval process that obtained 100% success rates of feasible solutions in each disruption instance. Overall, $CSREP_r$ retrieval was a robust operator. To ensure quality change of schedule retrieval, non-radical change and radical change were introduced and implemented. Besides that, $CSREP_r$ also repaired a disrupted schedule with a few changed cells quickly.

CHAPTER SIX

CONCLUSIONS AND RECOMMENDATIONS

This chapter summarizes the achievement of all five objectives, highlights the contributions to the body of knowledge on hybrid EA and its benefits to the nurse management. Finally, limitations of the present research and suggestions for future work are highlighted.

6.1 Summary of Nurse Scheduling and Rescheduling Problem

A nurse scheduling and rescheduling problem (NSRP) looks into the matters of nurse capacity, preferences and uncertainty in order to run a ward effectively. To tackle NSRP, the hybridization of evolutionary algorithm (EA) and cuckoo search (CS) was employed to utilize the available nurses by considering high nurse preference, fairness, and quality and quantity changes of disruption. Preference was sustained even in rescheduling process. Also, the integration of scheduling and rescheduling was considerably explored.

6.1.1 Achievement of Research Objectives

All five objectives were successfully met in solving a nurse scheduling and rescheduling problem by minimizing constraint violations. The first objective was met by reviewing the relevant literature and expert opinions, all relevant constraints and parameters that made up the rules and nurses preference in relation to nurse skills and staffing size were considered. In the second objective, adjustment or change for giving low impact to other nurses in rescheduling problem were discussed and accomplished in Section 4.3.2 and Section 5.6. The principles of pre-retrieval and retrieval were particularly reported in Section 4.3.2.9. As a whole, nurse coverage and high nurse preferences were not merely achieved in scheduling but also in rescheduling. The aspect of pursuing little schedule disruption with quality retrieval, fair on-call delegation, and highly desired nurse preferences such as Integrated Requested Off days were all involved. In that, a win-win situation's schedule was produced which attained nurse contractual request and nurse personal request.

Enhancing the operators of evolutionary algorithm in parent selection and crossover was stated in the third and fourth objective. In the third objective, Maximax and Maximin parent selection, Discovery Rate parent selection, and Discovery Rate Tournament parent selection were constructed as the newly modified parent selection operators to acclimatize population diversification. This third objective was accomplished with detailed procedures presented in Section 4.5.4 and the implementation result in Section 5.3.

The fourth objective was accomplished with detailed procedures presented in Section 4.5.5. It was to construct newly modified crossover operators for the scheduling problem and presented as a repair operator for the rescheduling problem. Several experiments of the newly crossover operators were tested in Section 5.4. Basically, the idea of Two-factor Blockwise crossover and Max[4x4]CSREP crossover were to enhance the flexibility of crossing over and enrich little exploitation element to avoid slow convergence. With these improvements, Max[4x4]CSREP was capable in transforming itself into a repair operator named $CSREP_r$.

In evaluating the performance of several evolutionary models and what-if analysis for NSRP, the last objective was achieved, as shown in Section 5.5 and Section 5.6. To conclude, hybridization of evolutionary algorithm and cuckoo search formed the basis for DrT_CSREP and $CSREP_r$. In that, they were promising in terms of model effectiveness, efficiency, accuracy, and reliability.

6.2 Research Contributions and Benefits

By using our enhanced cooperative hybridization (i.e., evolutionary algorithm with cuckoo search), the amalgamation of nurse scheduling and rescheduling problem (NSRP) can be solved, hence enhancing nurse management. Hence, the present research contributes mainly to EA's body of knowledge and nurse management system. Consequently, several groups of people can be benefited from the findings of this research. They are the head nurses, nurses, and patients.

6.2.1 Contribution to the Body of Knowledge

Essentially, EA's exploration and exploitation are having a contradictory nature. Each of them resulting to the extreme act of convergence, either it is fast or slow. Hence, EA intends to commit on a better balancing point to these principles. For this reason, our EA hybridization had watched over the population diversification, selective pressure, randomization principle and convergence issue through enhancing parent selection operator and crossover operator.

Mainly, to the best of our knowledge, this research is the first to construct a hybridization of evolutionary algorithm with cuckoo search in coorperative architecture to solve the integration of nurse scheduling and rescheduling problem.

The enhanced EA-based hybridization technique was found to be flexible in its search with numerous constraints involved. Based on the outstanding result of D_rT_CSREP model, this hybrid evolutionary algorithm with cuckoo search technique was said to have balanced the contradictory nature of exploration and exploitation successfully. The accomplishment has highlighted the capability of modern heuristics in hybridization, thought they are simple.

Next, two newly modified matrix crossover operators were created and named as Two-factor Blockwise crossover (2F Blockwise) and Cuckoo Search Restriction Enzyme Point crossover (Max[4x4]CSREP). They contributed to achieving a more flexible way of crossing over and enhancing permutation, hence, enlarging the exploration search. In addition to cuckoo search, the restriction enzyme point was a concept adopted from the principle of microbiology on DNA. The cooperative cuckoo search with Restriction Enzyme Point was firstly integrated in the proposed crossover operator (i.e., Max[4x4]CSREP). One noticeable point is that the superiority of hybridizing Cuckoo Search in CSREP crossover operator was able to produce best-so-far offspring for the scheduling problem and a better fit retrieved solution for the rescheduling problem. Overall, CSREP is used to enrich little exploitation to an over exploration search in order to evade slow convergence matter.

Three newly modified parent selection operators were created in this research. They were Maximax and Maximin parent selection, Discovery Rate parent selection, and Discovery Rate Tournament parent selection. These dissimilarity relationship selections were created to manipulate population diversity, without adding any filtering heuristics to an initial population. Essentially, the selections concerned some 'rare search area' throughout a population as well as prevented premature convergence. In all, this research found that elite parents and relative difference between them were vital elements in designing parent selection operator.

With regards to population size, small size of population was said not favorable because might lost population's diversity. In fact, this research showed that smaller population size could benefit to a problem that is low permutation in an initial population which restricted by numerous of constraints. In this case, large population size may take a risk of producing quite similar initial individuals (i.e., low diversity). Note that, one stipulation is that flexibility is needed in order to reduce the risk of getting premature convergence, which means small size population with flexible search in selection operator could also cover a solution space effectively.

Lastly, this research introduced a systematic way in settinga penalty value to each level of constraint violation (i.e. hard, semi-hard, and soft constraint). Particular to the semi-hard constraint, it can be a linkage of hard and soft constraint whereby effectively handling the integrated scheduling and rescheduling problem. Fitness calculation was able to prioritize the constraints. Therefore, by examining the penalty value, it is able to direct operators to tackle certain level of constraints violation.

6.2.2 Contribution to the Nurse Management

Based on the enhancement of fitness calculation, this research was able to handle a vast number of constraints by determining their priority as well as the complexity level of the constraints. The semi-hard constraint strategy was introduced to handle the nurse coverage and nurse preference in the scheduling and rescheduling problem.

Uncertain absenteeism that had resulted in different levels of shortage severity suggests that the handling of the disruption could be managed by considering the different levels of coverage and nurse preference.

The combination of nurse scheduling and rescheduling is the advancement to one part of nurse management system. This model (i.e., the integrated scheduling and rescheduling model) can be helpful in adjusting the demand for nurses' services in light of disruptions or without disruptions. The model also fully utilizes the nurses by considering their availability and preferences to improve their productivity and services. This integrated scheduling and rescheduling model has been stepping further ahead to befit to real-world condition.

This model strictly considers shift continuity arrangements when retrieving a schedule using the cyclical rhythm principle involving past duty, current duty, and future duty. For example, nurses who already did the morning shift duty were excluded when the disruption happens in the evening shift of the same day. The impact of changing current shift that could affect future duty was considered as well. Essentially, this complexity was solved when scheduling and rescheduling were integrated and resulting a trustable schedule. In the previous studies, these types of constraints were lack considered in scheduling stage alone although scheduling and rescheduling stages face understaffing problem. This is because scheduling is a planning process; it only subjects to postulating a schedule, yet rescheduling is an implementing process; its point of view only subjects to current reality real-time issue. Therefore, the separation stages were hard to mull over the continuity arrangements of past, current and future shifts.

Nurse fairness and preferences elements which intended to reduce internal conflict amongst nurses, were included in the scheduling and rescheduling model. This accomplishment has resulting a better chances of having significant Integrated Request Off day among nurses; equivalent delegation of on-call nurse among the nurses of a ward; equivalent chances of having significant weekend off day among nurses; and equal balance number of Morning shift and Evening shifts for each nurse. Whereby, the fairness of average number of nurses for each type of shifts was advocated. To note, the first two preferences (i.e., Integrated Request Off and equivalent on-call delegation) were newly implemented in nurse scheduling problem. They were classified as highly desired preferences in our NSRP.

Our rescheduling was able to provide contingency arrangements that were pre-retrieval and retrieval alternatives in light of the risk (i.e., level of disruption) due to uncertainty. The pre-retrieval scrutinized the seriousness of a disruption based on a schedule readiness. Pre-retrieval could examine the current nurse coverage condition and hard constraints violation whereby readjusted the current absent nurse's schedule or suggest other nurses who are assigned off duty but still willing to assist if an emergency cases happened on that day. Certainly, head nurse must have good relationship with the nurses to gain nurses' willingness and corporation; that could be earned through the success of fulfilling highly desired preferences. Essentially, this rescheduling was to preserve the original schedule when the schedule disruption was not tremendously serious.

In a non-survival condition which a ward could not operate with serious understaffing problem, this integrated model offers more than one choice of retrieval strategy. There are two considerations which are seeking higher quality retrieved schedule and lower quantity of changes. Retrieval operator could recreate a new schedule accordingly (i.e., retrieval but with radically restructuring the remaining days) or retrieve with few adjustments (i.e., retrieval but no radical change to keep the original schedule in tact as much as possible). By then, quality change and quantity change were both taking into consideration automatically during rescheduling.

The compensation off-day for nurses who were assigned the on-call duty was completed in our model. To the best of our knowledge, this has not been implemented in a nurse rescheduling model. Despite giving the compensation off at the right time, the model also attempts to adjust the off days in a consecutive manner. This is meant for comforting the on-call nurses by giving them significant rest times as well as showing gratitude for their cooperation.

Last but not the least, this computerized model is flexible enough to be modified to suit other industries. By including more detailed data estimation, a robust model for manufacturing manpower scheduling, airline task-based crew recovery problem, and railway tour-of-duty rescheduling problem can also be constructed.

6.2.3 Head Nurses' Benefit

For the head nurse, most of them create a schedule manually, therefore scheduling and rescheduling is time consuming. The integration of nurse scheduling and rescheduling model aids the head nurse in responding to sudden changes to nurse allocation. This human-like model also improves fair distribution of nurses' assignment and advocate of humanitarian management (e.g., tolerance between head nurse and nurses). When the work environment is perceived as fair and desirable, the nurses' intention to leave the organization may be reduced.

6.2.4 Nurses' Benefit

An efficient and effective management of manpower through scheduling and rescheduling can enhance nurse engagement to their job. This is because the nurses can meet their work preferences in terms of on-duty and off-duty assignment. Particularly, those highly desired preferences (e.g., timely requested off, weekend off, and consecutive compensation off days) and trusted/predicted schedule can be achieved. By doing so, the nurses will be able to balance their work and life simultaneously, hence reducing internal conflict, and enhance their health as well as quality marital life.

6.2.5 Patients' Benefit

When there are readily available nurses, better patient safety can be achieved. In addition, because the schedule is prepared fairly and by considering their preferences, the work environment is enhanced. So, when they are satisfied with such arrangement, this spills over to the patients' safety and recovery.

6.3 Research Limitations

Patient safety can be more assured if nurse task assignment is taken into account. Nurse task assignment involves matters of workload balance which regards to equal nursing time provided to each patients as well as same number of patients that need to be taking care of. In fact, the on floor nursing care is complicated to be caught up in the research due to the variation of patient health condition in which mainly compromises with doctor's diagnosis. For that reason, shift continuity elements in nurse task assignment (e.g., particular nurse is required to take up shift due to particular patient care) and patient classification system are excluded in this research. The nurse coverage in our research was determined from on the data gathered. However, another alternative is to deal with patient classification system, which classifies patients into several categories based on the severity of their illness. Therefore, more precise nurse demand and their allocation can be estimated during rescheduling since the patients' condition changes from time to time. Perhaps, search technique may not be a well fit gadget to support this data inputting matter.

In this research, we excluded temporary nurses, such as, part-time nurses, volunteer helpers, agency nurses and etc. who work to fulfill the shortage within a short period of time. This group of nurses was excluded because the use of such nurses is not commonly practiced in Malaysia due to financial limitation.

6.4 Future Work

For future work, some suggestions are offered in light of the limitations highlighted above.

In order to manage the manpower effectively in health care organizations, it is suggest that an effective patient information system is developed to support the nurse management system. For example, information such as the condition of the patient, the duration of his/her medical treatment and bed capacity should be made available.

In doing so, determining accurately the level of the nurses' skills can be achieved, hence effectively deploying human resource.

In addition, some subjective data such as nurse's experiences and expertise, personal goals, and a mixture of nurses in a shift can be considered in NSRP specifically in on-call allocation. By doing so, better teamwork and thus better work environment can be achieved. In this way, the decision support system can be developed in such a way to mimic a real problem.

Even though penalty value is used to evaluate fitness to satisfy constraints, one may fall short of fair techniques comparisons amongst scholars because of different weighted values. To this matter, future work can take remedial measures against the variation of fitness evaluation such as setting a standard benchmark by distance.

The hybridization search techniques are the most commonly used and effective approach to nurse scheduling. In this research, the simplicity of cuckoo search had drawn us to this cooperative architecture. Nevertheless, there are some others modern heuristic (e.g. African Wild Dog Algorithm) can be explored as well as compared with our cuckoo search integration model. We hope that this research may shed light to the modern heuristic hybridization.

Integrating cuckoo search at crossover operator was shown to be a viable hybridization in solving both the scheduling and rescheduling problem. Its ability may further improve evolutionary algorithm by applying it to the parent selection operator in larger population sizes. Besides, it is suggested that cuckoo search is integrated into the direct mutation operator since exploration and exploitation search can be flexibly adjusted by considering some constraints.

REFERENCES

Abdallah, K. S., & Jang, J. (2014). An exact solution for vehicle routing problems with semi-hard resource constraints. *Computers and Industrial Engineering*, *76*, 366-377. doi: 10.1016/j.cie.2014.08.011

Abdul-Rahman, S., Burke, E. K., Bargiela, A., McCollum, B., & Özcan, E. (2014). A constructive approach to examination timetabling based on adaptive decomposition and ordering. *Annals of Operations Research, 218*(1), 3-21.

Abdul-Rahman, S., Sobri, N. S., Omar, M. F., Benjamin, A. M., & Ramli, R. (2014). Graph coloring heuristics for solving examination timetabling problem at universiti utara malaysia. In *AIP Conference Proceedings of the 3rd International Conference on Quantitative Sciences and Its Applications (ICOQSIA 2014), Langkawi Malaysia, 1635*, 491-496.

Abraham, A., & Ramos, V. (2003). Web usage mining using artificial ant colony clustering and genetic programming. In *IEEE Congress on Evolutionary Computation (CEC 2003), Australia*, 1384-1391.

AbuAlRub, R. F. (2004). Job stress, job performance, and social support among hospital nurses. *Journal of Nursing Scholarsh*, *36*(1), 73-8.

Abu-Srhahn, A. A., & Al-Hasan, M. (2015). Hybrid Algorithm using Genetic Algorithm and Cuckoo Search Algorithm for Job Shop Scheduling Problem. *International Journal of Computer Science Issues (IJCSI)*, *12*(2), 288.

Abu-Srhan, A. A., & Al Daoud, E. (2013). A hybrid algorithm using a genetic algorithm and cuckoo search algorithm to solve the traveling salesman problem and its application to multiple sequence alignment. *Internatioanl Journal of Advanced Science and Technology*, *61*, 29-38.

Adamopoulos, K., Harman, M., & Hierons, R. M. (2004). How to overcome the equivalent mutant problem and achieve tailored selective mutation using co-evolution. In Deb, K. (Ed), *Genetic and Evolutionary Computation (GECCO)*, 1338-1349. Springer Berlin Heidelberg.

Ahmad-Beygi, S., Cohn, A., & Lapp, M. (2010). Decreasing Airline Delay propagation By Re-Allocating Scheduled Slack. *IIE Transactions*, *42*(7),478-489, 2010.

Ahmed, A. & Alkhamis, M. (2008). Simulation optimization for an emergency department healthcare unit in kuwait. *European Journal of Operational Research*, *198*, 936-942

Aickelin, U., & Dowsland, K. A. (2000). Exploiting problem structure in a genetic algorithm approach to a nurse rostering problem. *Journal of Scheduling, 3*(3), 139-153.

Aickelin, U., & Dowsland, K. A. (2004). An Indirect Genetic Algorithm for a Nurse Scheduling Problem. *Computers and Operations Research, 31*(5), 761-778.

Aickelin, U., & Li, J. (2004). The application of bayesian optimization and classifier systems in nurse scheduling. In *Proc. of the 8th International Conference on Parallel Problem Solving from Nature.* (PPSN VIII), Lecture Notes in Computer Science, *3242*, 581-590.

Aickelin, U., & White, P. (2004). Building better nurse scheduling algorithms. *Annals of Operations Research, 128*, 159-177.

Aickelin, U., Burke, E. K., & Li, J. P. (2007). An estimation of distribution algorithm with intelligent local search for rule-based nurse rostering. *Journal of the Operational Research Society*, 1-21.

Aiken, L. H. (2010). The California nurse staffing mandate: implications for other states. *Issue Brief: Leonard Davis Institute of Health Economics*, *15*(4), 1-4.

Aiken, L. H., Clark, S. P., Sloane, D. M., Sochalski, J., & Silber, J. H. (2002). Hospital nurse staffing and patient mortality, nurse burnout, and job dissatisfaction. *Journal of the American Medical Association, 288*(16), 1987-1993.

Aiken, L. H., Clarke, S. P., Sloane, D. M., Lake, E. T., & Cheney, T. (2009). Effects of hospital care environment on patient mortality and nurse outcomes. *Journal of Nursing Administration*, *39*(7), 45-51.

Aiken, L. H., Clarke, S. P., Sloane, D. M., Sochalski, J., & Silber, J. H. (2002). Hospital nurse staffing and patient mortality, nurse burnout and job dissatisfaction. *The Journal of the American Medical Association*, *288*(16), 1987-1993.

Akihiro, K., Chika, Y., & Hiromitsu, T. (2005). Basic solutions of nurse scheduling problem using 3d-structured binary neural networks. *Transactions of Information Processing Society of Japan*, *46*, 41-47.

Al-Ahmadi, H. (2009). Factors affecting performance of hospital nurses in riyadh region, saudi arabia. *International Journal of Health Care Quality Assurance*, *22*(1). 40-54.

Al-Dulaimi, B. F., & Ali, H. A. (2008). Enhanced traveling salesman problem solving by genetic algorithm technique (tspga). *World Academy of Science, Engineering And Technology*, *38*, 296-302.

Al-Naqi, A., Erdogan, A. T., & Arslan, T. (2010). Fault tolerance through automatic cell isolation using three-dimensional cellular genetic algorithms. In *2010 IEEE Congress on Evolutionary Computation (CEC)*. New York: IEEE, 1-8.

Ashlock, D. (2005). *Evolutionary computation for modeling and optimization*. USA: Springer.

Augusto, V., Xie, X., & Perdomo, V. (2010). Operating theatre scheduling with patient recovery in both operating rooms and recovery beds. *Computers and Industrial Engineering*, *58*(2), 231-238.

Aytug, H., & Koehler, G. J. (2000). New stopping criterion for genetic algorithms. *European Journal of Operational Research*, *126*(3), 662-674.

Azaiez, M. N., & Al Sharif, S. S. (2005). A 0-1 goal programming model for nurse scheduling. *Computers and Operations Research*, *32*, 491-507.

Back, T., & Schwefel, H. P. (1993). An overview of evolutionary algorithms for parameter optimization. *Evolutionary Computation*, *1*(1), 1-23.

Back, T., Hammel, U., & Schwefel, H. P. (1997). Evolutionary computation: comments on the history and current state. *IEEE Transactions on Evolutionary Computation*, *1*(1), 3-17.

Bai, R. Burke, E. K., Kendal, G., Li, J. P., & McCollum, B. (2007). A hybrid evolutionary approach to the nurse rostering problem. *IEEE Transaction on Evolutionary Computation*, 1-10.

Bailey, R. N., Garner, K. M., & Hobbs, M. F. (1997). Using simulated annealing and genetic algorithms to solve staff-scheduling problems. *Asian-Pacific Journal of Operational Research, 14*(2), 27-44.

Baker, K. R. (1976). Workforce allocation in cyclical scheduling problems: a survey. *Operational Research Quarterly*, *27*(1), 155-167.

Ballester, P. J., & Carter, J. N. (2005). An Effective Real-Parameter Genetic Algorithms with Parent Centric Normal Crossover for Multimodal Optimization. In *Genetic and Evolutionary Computation Conference*, *Lecture Notes in Computer Science*. Springer. *3102*, 901-913.

Banyal, S. (2011, June 22). Nurse pov forums: nurses on call. Retrieved from http://community.advanceweb.com/forums/thread/3103.aspx

Bard, J. F. (2004a). Selecting the appropriate input data set when configuring a permanent workforce. *Computers and Industrial Engineering*, *47*(4), 371-389.

Bard, J. F. (2004b). Staff scheduling in high volume service facilities with downgrading. *IIE Transactions*, *36*(10), 985-997.

Bard, J. F., & Purnomo, H. W. (2004b). Real-time scheduling for nurses in response to demand fluctuations and personnel shortages. In *Proc. of 5th International Conference on the Practice and Theory of Automated Timetabling,* Pittsburgh, 67-87.

Bard, J. F., & Purnomo, H. W. (2005a). Hospital-wide reactive scheduling of nurses with preference considerations. *IIE Transactions*, *37*, 587-608.

Bard, J. F., & Purnomo, H. W. (2005b). Short-term nurse scheduling in response to daily fluctuations in supply and demand. *Health Care Management Science. 8*, 315-324.

Bard, J. F., & Purnomo, H. W. (2005c). A column generation-based approach to solve the preference scheduling problem for nurses with downgrading. *Socio-Economic Planning Sciences, 39*, 193-213.

Bard, J. F., & Purnomo, H. W. (2007). Cyclic preference scheduling of nurses using a lagrangian-based heuristic. *Journal of Scheduling*, *10*, 5-23.

Barnett, T., Namasivayam, P., & Narudin, D. A. A. (2010), A critical review of the nursing shortage in Malaysia. International Nursing Review, *57*(1), 32-39. doi: 10.1111/j.1466-7657.2009.00784.x

Barrangou, R., Fremaux, C., Deveau, H., Richards, M., Boyaval, P., Moineau, S., ... & Horvath, P. (2007). CRISPR provides acquired resistance against viruses in prokaryotes. *Science*, *315*(5819), 1709-1712.

Baumelt, Z., Dvorak, J., Sucha, P., & Hanzalek, Z. (2013). An acceleration of the algorithm for the nurse rerostering problem on a graphics processing unit. In *Proceedings of 5th International Conference on Applied Operational Research, Lecture Notes in Management Science, 5*, 101-110.

Baumelt, Z., Sucha, P., & Hanzalek, Z. (2007). Nurse scheduling web application. In *Proceedings of the 26th Workshop of the UK Planning and Scheduling Special Interest Group. Prague, UK MFF.*

Beddoe, G. R., & Petrovic, S. (2006). Selecting and weighting features using a genetic algorithm in a case-based reasoning approach to personnel rostering. *European Journal of Operational Research, 175*, 649-671.

Beddoe, G. R., & Petrovic. S. (2007). Enhancing case-based reasoning for personnel rostering with selected tabu search concepts. *Journal of the Operational Research Society*, *58*(12), 1586-1598.

Beddoe, G. R., Petrovic, S., & Berghe, G. V. (2002). Storing and adapting repair experiences in employee rostering. *Selected Papers of the 4th International. Conference on Practice and Theory of Automated Timetabling*, *2740*, 148-165.

Beddoe, G., & Petrovic. S. (2007). Enhancing case-based reasoning for personnel rostering with selected tabu search concepts. *Journal of the Operational Research Society*, *58*(12), 1586-1598.

Belian, J. (2006). Exact and heuristic methodologies for scheduling in hospitals-problems, formulations and algorithms. *4OR: A Quarterly Journal of Operations Research*, *5*(2), 157-160.

Belian, J., & Demeulemeester, E. (2006). Scheduling trainees at a hospital department using a branch-and-price approach. *European Journal of Operational Research, 175*, 258-278.

Belian, J., & Demeulemeester, E. (2008). A branch-and-price approach for integrating nurse and surgery scheduling. *European Journal of Operational Research, 189*, 652-668.

Bellanti, F., Carello, G., Croce, F. D., & Tadei, R. (2004). A greedy-based neighborhood search approach to a nurse rostering problem. *European Journal of Operational Research, 153*, 28-40.

Berghe, G. V. (2002). *An advanced model and novel meta-heuristic solution methods to personnel scheduling in healthcare.* PhD dissertation, University of Gent.

Bernama. (2011). Staff shortage makes nursing jobs harder. Retrieved on September 15, 2012 from http://www.theborneopost.com/2011/12/01/staff-shortage-makes-nursingjobsharder/

Berrada, I., Ferland, J. A., & Michelon, P. (1996). A multi-objective approach to nurse scheduling with both hard and soft constraints. *Socio-Economic Planning Sciences, 30*(3), 183-193.

Berry, A., & Vamplew, P. W. (2004). Pod can mutate: a simple dynamic directed mutation approach for genetic algorithms. In *Proceedings of AISAT2004: International Conference on Artificial Intelligence in Science and Technology.* University of Tasmania, 200-205.

Bester, M. J., Nieuwoudt, I., & Vuuren, J. H. V. (2007). Finding good nurse duty schedules: a case study. *Journal of Scheduling. 10*, 387-406.

Bhadury, J., & Radovilsky, Z. (2006). Job rotation using the multi-period assignment model. *International Journal of Production Research, 44*(20), 4431-4444.

Billings, J. S. (1985). *A comparision of roles of head nurses according to hospital size.* University of Minnesota.

Blum, C., & Roli, A. (2003). Metaheuristics in combinatorial optimization: overview and conceptual comparison. *ACM Computing Survey, 35*(3), 268-308.

Bożejko, W., & Wodecki, M. (2005). A hybrid evolutionary algorithm for some discrete optimization problems. *IEEE Computer Society*, 325-331.

Bradley, D. J., & Martin, J. B. (1990). Continuous personnel scheduling algorithms: a literature review. *Journal of Society for Health Systems, 2*, 8-23.

Brajevic, I., Tuba, M., & Bacanin, N. (2012). Multilevel image thresholding selection based on the cuckoo search algorithm. *In Proceedings of the 5th International Conference on Visualization, Imaging and Simulation,* Sliema, Malta, 217-222.

Bratu, S., & Barnhart, C., (2006). Flight operations recovery: new approaches considering passenger recovery. *Journal of Scheduling*, *9*, 286-298.

Bremermann, H. J. (1958). *The evolution of intelligence. the nervous system as a model of its environment.* Technical Report 1, Contract 477(17), Department of Mathematics, university of Washington, Seattle.

Briesacher, B. A., Field, T. S., Baril, J., & Gurwitz, J. H. (2009). Pay-for-performance in nursing homes. *Health Care Financing Review*, *30*(3), 1-13.

Brucker, P., Burke, E. K., Curtois, T., Qu, R., & Berghe, G. V. (2010). A shift sequence based approach for nurse scheduling and a new benchmark dataset. *Journal of Heuristics*, *16*(4), 559-573.

Brucker, P., Burke, E. K., Curtois, T., Qu, R., & Berghe, G. V. (2007). *Adaptive construction of nurse schedules: a shift sequence based approach.* Computer Science Technical Report. University of Nottingham.

Brucker, P., Qu, R., Burke, E., & Post, G. (2005). A decomposition, construction and postprocessing approach for nurse rostering. In *Multidisciplinary International Conference on Scheduling: Theory and Applications*, New York, 397-406.

Brusco, M. J., & Showalter, M. J. (1993). Constrained nurse staffing analysis. *Omega, International Journal of Management Sciences*, *21*(2), 175-186.

Brusco, M., & Jacobs, L. (2001). Starting-time decisions in labor tour scheduling: an experimental analysis and case study. *European Journal of Operational Research*, *131*, 459-475.

Buchan, J., & Calman, L. (2004). *The global shortage of registered nurses: an overview of issues and actions.* International Council of Nurses. Geneva, Switzerland.

Bureerat, S., & Sriworamas, K. (2013). Simultaneous topology and sizing optimization of a water distribution network using a hybrid multiobjective evolutionary algorithm. *Applied Soft Computing*, *13*(8), 3693-3702.

Burke, E. K., & Smith, A. J. (2000). Hybrid evolutionary techniques for the maintenance scheduling problem. *The IEEE Transactions on Power System*, *15*(1), 122-128.

Burke, E. K., De Causmaecker, P., Berghe, G. V., & Landechem, H. V. (2004). The state of the art of nurse rostering. *Journal of Scheduling*, *7*, 441-499.

Burke, E. K., De Causmaecker, P., Petrovic, S., & Berghe, G. V. (2003). Variable neighbourhood search for Nurse rostering problems. In *Metaheuristics: Computer Decision-Making Chapter 7*, Kluwer, 153-172.

Burke, E. K., De Causmaecker, P., Petrovic, S., & Berghe, G. V. (2001). Fitness evaluation for nurse scheduling problems. In *Proceedings of the 2001 Congress on Evolutionary Computation*, 2, 1139-1146.

Burke, E. K., De Causmaecker, P., Petrovic, S., & Berghe, G. V. (2002). A multi criteria meta-heuristic approach to nurse rostering. In *Proceedings of World on Congress on Computational Intelligence*, 1197-1202.

Burke, E. K., Cowling, P., Caumaecker, P. D., & Berghe, G. V. (2001). A memetic approach to the nurse rostering problem. *Applied Intelligence Special Issue on Simulated Evolution and Learning, 15*(3), 199-214.

Burke, E. K., Curtois, T., Post, G., Qu, R., & Velman, B. (2008). Hybrid heuristic ordering and variable neighbourhood search for the nurse rostering problem. *European Journal of Operational Research*, 330-341.

Burke, E. K., Curtois, T., Qu, R., & Berghe, G. V. (2010). A scatter search for the nurse rostering problem. *Journal of the Operational Research Society, 61* (11), 1667-1679.

Burke, E. K., Kendall, G., & Soubeiga, E. (2003). A tabu search hyperheuristic for timetabling and rostering. *Journal of Heuristics*, *9*(6), 451-470.

Burke, E. K., Mccollum, B., Meisels, A., Petrovic, S., & Qu, R. (2007). A Graph-based hyper-heuristic for educational timetabling problems. *European Journal of Operational Research*, *176*, 177-192.

Burke, E. K., Petrovic, S., & Qu, R. (2006). Case-based heuristic selection for timetabling problems. *Journal of Scheduling*, *9*(2), 115-132.

Burke, E., & Soubeiga, E. (2003). A real-world workforce scheduling problem in the hospitality industry: theoretical models and algorithmic methods. *In Proc. of the 3rd EU/ME Workshop on real world applications of metaheuristics*. Antwerp, Belgium.

Cai, X., & Li, K. N. (2000). A genetic algorithm for scheduling staff of mixed skills under multi-criteria. *European Journal of Operational Research*, *125*, 359-369.

Cheang, B., Li, H., Lim, A., & Rodrigues, B. (2003). Nurse rostering problems- a bibliographic survey. *European Journal of Operational Research*, *151*, 447-460.

Chen, J. G., & Yeung, T. (1992). Development of a hybrid expert system for nurse shift scheduling. *Journal of Industrial Ergonomics, 9*(4), 315–328.

Cheng, B. M. W., Lee, J. H. M., & Wu, J. C. K. (1997). A nurse rostering system using constraint programming and redundant modeling. IEEE Transactions on Information Technology in Biomedicine, *1*(1), 44-54.

Chiaramonte, M. V. (2008). *Competitive nurse rostering and rerostering*. Arizone State University.

Chipas, A., & McKenna, D. (2011). Stress and burnout in nurse anesthesia. *Journal of the American Association of Nurse Anesthetists, 79*(2), 122-128.

Choy, M., & Cheong, M.L.F. (2012). A flexible mixed integer programming framework for nurse scheduling. In Proceedings of computing Research Repository (CoRR). *arXiv preprint arXiv:1210.3652*

Clark, A. R., & Walker, H. (2011). Nurse rescheduling with shift preferences and minimal disruption. *Journal of Applied Operational Research, 3*(3), 148-162.

Clausen, J., Hansen, J., Larsen, J., & Larsen, A. (2001). *Disruption management-operations research between planning and execution.* (Technical Report IMM-REP-2001-15. IMM). Technical University of Denmark.

Cohen, A., & Golan, R. (2007). Predicting absenteeism and turnover intentions by past absenteeism and work attitudes: an empirical examination of female employees in long term nursing care facilities. *Career Development International, 12*(5), 416-432.

Cohen, J., Stuenkel, D., & Nguyen, Q. (2009). Providing a healthy work environment for nurses: the influence on retention. *Journal of Nursing Care Quality, 24*(4), 308-315.

Coleman, R.M. (1996). The efficiency is in the schedule. *HR Focus, 73*(10), 17-18.

Cowling, P, Kendall, G., & Soubeiga, E. (2002). Hyperheuristics: a robust optimization method applied to nurse scheduling. *Journal of Heuristics*, 851-860.

Danciu, D. (2003). Evolutionary timetabling using biased genetic operators. *Journal of Computing and Information Technology, 11*(3), 193-199.

De Causmaecker, P., Demeester, P., & Berghe, G. V. (2009). A decomposed metaheuristic approach for a real-world university timetabling problem. *European Journal of Operational Research, 195*(1), 307-318.

Deep, K., & Thakur, M. (2007). A new crossover operator for real coded genetic algorithms. *Applied Mathematics and Computation, 188*, 895-911.

Deep, K., Singh, K. P., Kansal, M. L., & Mohan, C. (2009). A real coded genetic algorithm for solving integer and mixed integer optimization problems. *Applied Mathematics and Computation, 212*(2), 505-518.

Dellasega, C. A. (2009). Bullying among nurses. *American Journal of Nursing, 109*(1), 52-58.

Dep, K. (2000). Introduction to selection. In *Evolutionary Computation I: Basic Algorithms and Operators (Back, T., Fogel, D. B., & Michalewics, Z., eds.).* 166-171, Bristol: Institute of Physics Publishing.

Dias, T. M., Ferber, D. F., Souza, C. C., & Moura, A. V. (2003). Constructing nurse schedules at large hospitals. *International Transactions in Operational Research, 10*, 245-265.

Dowsland, K. (1998). Nurse scheduling with tabu search and strategic oscillation. *European Journal of Operational Research, 106*, 393-407.

Dowsland, K., & Thompson, J. (2000). Solving a nurse scheduling problem with knapsacks, networks and tabu search. *Journal of the Operational Research Society, 51*, 825-833.

Dunton, N., Gajewski, B., Klaus, S., & Pierson, B. (2007).The relationship of nursing workforce characteristics to patient outcomes.*The Online Journal of Issues in Nursing*, *12*(3), 3. doi: 10.3912/OJIN.Vol12No03Man03

Dutta, A. (1990). Reacting to scheduling exceptions in FMS environments. *IIE Transactions*, *22*(4), 300-314.

Eastaugh, S. R. (2007). Hospital nurse productivity enhancement. *Journal of Health Care Finance*, *33*(3), 39-47.

Eiben, A. E., & Smith, J. E. (2003). *Introduction to evolutionary computing*. New York: Springer.

Eiben, A. E., Van Kemenade, C. H. M., & Kok, J. N. (1995). Orgy in the computer: multi-parent reproduction in genetic algorithms. In *Proceedings of the 3rd European Conference on Artificial Life,* 929, 934-945. Springer-Verlag.

Engku Muhammad Nazri, E. M. N. (2001). *A heuristic to schedulng security personnel at local universities in malaysia.* Unpublished Ph.D. Thesis. Universiti Utara Malaysia.

Ernst, A. T., Jiang, H., Krishnamoorthy, M., & Sier, D. (2004). Staff scheduling and rostering: a review of applications, methods and models. *European Journal of Operational Research*, *153*, 3-27.

Eshelman, L. J., Caruana, R. A. & Schaffer, J. D. (1989). Biases in the crossover landscape. In *Proceedings of the Third International Conference on Genetic Algorithms*. San Mateo, CA: Morgan Kaufmann Publishers. 10-19.

Eskandari, H., & Geiger, C. D. (2008). A fast pareto genetic algorithm approach for solving expensive multiobjective optimization problems. *Journal of Heuristics*, *14*(3), 203-241.

Fletcher, C. E. (2001). Hospital rns' job satisfactions and dissatisfactions. *Journal of Nursing Administration*, *31*(6), 324-31.

Flynn, W. J., Mathis, R. L., & Jackson, J. H. (2007). *Healthcare human resource management* (2nd Edition). USA: Thomson South-Western.

Fogel, D. B. (1995). *Evolutionary Computation: Toward a New Philosophy of Machine Intelligence*. Piscataway, NJ: IEEE Press.

Ford, S. (2012, March). Mandatory staffing ratios aid recruitment and retention. *Nursing Times*. Retrieved from http://www.nursingtimes.net/nursing-practice/clinical-zones/management/mandatory-staffing-ratios-aid-recruitment-and-retention/5042815.article

Ford, S. (2013). Exclusive: nurses feeling under pressure, understaffed and undervalued. *Nursing Times*. Retrieved from http://www.nursingtimes.net/nursing-practice/clinical-zones/management/exclusive-nurses-feeling-under-pressure-understaffed-and-undervalued/5063786.article

Fung, S. K. L., Leung, H. F., & Lee, J. H. M. (2005). Guided complete search for nurse rostering problems. *In Proc. of the 17th IEEE International Conference on Tools with Artificial Intelligence, Computer Society*, 1082-3409.

Garcia-Martinez, C., Lozano, M., Herrera, F., Molina, D., & Sanchez, A. M. (2008). Global and local real-coded genetic algorithms based on parent-centric crossover operators. *European Journal of Operational Research, 185*, 1088-1113.

Garrett, D., & McDaniel, A. (2001). A new look at nurse burnout: the effects of environmental uncertainty and social climate. *Journal of Nursing Administration, 31*(2), 91-96.

Glass, C. A., & Knight, R. A. (2010). The nurse rostering problem: a critical appraisal of the problem structure. *European Journal of Operational Research, 202*(2), 379-389.

Goel, A., Archetti, C., & Savelsbergh, M. (2012). Truck driver scheduling in australia. *Computers and Operations Research, 39*, 1122-1132.

Goodman, M. D., Dowsland, K. A., & Thompson, J. M. (2009). A grasp-knapsack hybrid for a nurse-scheduling problem. *Journal of Heuristics, 15*, 351-379. doi 10.1007/s10732-007-9066-7

Gormley, D. K. (2011). Are we on the same page? staff nurse and manager perceptions of work environment, quality of care and anticipated nurse turnover. *Journal of Nursing Management, 19*, 33-40.

Grandoni, F., Könemann, J., Panconesi, A., & Sozio, M. (2005). Primal-dual based distributed algorithms for vertex cover with semi-hard capacities. In *Proceedings of the twenty-fourth annual ACM symposium on Principles of distributed computing, ACM,* 118-125.

Grefenstette, J. J. (1986). Optimization of control parameters for genetic algorithms. *IEEE Transactions On Systems, Man, And Cybernetics*, *16*(1), 122-128.

Grosan, C., & Abraham, A. (2007). Hybrid evolutionary algorithm: Methodologies, architectures, and reviews. *Hybrid Evolutionary Algorithms*, *75*, 1-17.

Grosan, C., Abraham, A., & Nicoara, M. (2005). Search optimization using hybrid particle sub-swarms and evolutionary algorithms. *International Journal of Simulation*, *6*(10-11), 60-79.

Gupta, B., & Dhingra, S. (2013). Analysis of genetic algorithm for multiprocessor task scheduling problem. *International Journal Of Advanced Research In Computer Science And Software Engineering*, *3*(7), 339-344.

Gutjahr, W. J., & Rauner, M. S. (2007). An aco algorithm for a dynamics regional nurse-scheduling problem in Austria. *Computers and Operations Research*, *34*, 642-666.

Hart, S. (2005). Adaptive heuristics. *Econometrica*, *73*(5), 1401-1430.

Haupt, R. L., & Haupt, S. E. (2004). *Practical genetic algorithms* (2nd Edition). New Jersey: John Wiley.

Hayes, L. J., O'Brien-Pallas, L., Duffield, C., Shamian, J., Buchan, J., Hughes, F., Spence Laschinger, H. K., North, N., & Stone, P. W. (2006). Nurse turnover: a literature review. *International Journal of Nursing Studies*, *43*, 237-263.

He, F., & Qu, R. (2012). A constraint programming based column generation approach to nurse rostering problems. *Computers and Operations Research*, *39*, 3331-3343.

Health Indicators. (2010). *Indicators for monitoring and evaluation of strategy health for all.* Retrieved on March 4, 2014 from http://www.moh.gov.my/images/gallery/publications/md/hi/hi_2010.pdf

Heinrich, J. (2001). Nursing workforce: emerging nurse shortages due to multiple factors. In *GAO report to health subcommittee on health.* i-15, Washington DC: United States General Accounting Office.

Heizer, J., & Render, B. (2006). *Operations management* (8th Edition). New Jersey: Prentice Hall.

Hertz, A. & Kobler, D. (2000). A framework for the description of evoluitionary algorithms. *European Journal of Operational Research*, *126*, 1-12.

Heylighen, F. (1992). A cognitive-systemic reconstruction of maslow's theory of self-actualization. *Behavioral Science*, *37*, 39-57.

Holland, J. H. (1992). *Adaptation in natural and artificial systems*. MIT Press, Cambridge.

Horio, M. (2005). A method for solving the 3-shift nurse scheduling problem through a general project scheduler based on the framework of RCPSP/τ.

Horrocks, N., & Pounder, R. (2006). *Working the night shift: preparation, survival and recovery: a guide for junior doctors.* Royal College of Physicians.

Horvath, P., & Barrangou, R. (2010). CRISPR/Cas, the immune system of bacteria and archaea. *Science, 327*(5962), 167-170.

Hsia, T. L., Lin, L. M., Wu, J. H., & Tsai, H. T. (2006). A framework for designing nursing knowledge management system. *Interdisciplinary Journal of Information, Knowledge, and Management. 1*, 13-22.

Huisman, D. (2007). A column generation approach for the rail crew re-scheduling problem. *European Journal of Operational Research, 180*, 163-173.

Hutter, M. & Legg, S. (2006). Fitness uniform optimization. *IEEE Transactions on Evolutionary Computation, 10*(5), 568-589.

Ikegami, A., & Niwa, A. (2003). A subproblem-centric model and approach to the nurse scheduling problem. *Mathematical Programming, 97*(3), 517-541.

Ingersoll, G. L., Olsan, T., Drewcates, J., DeVinney B. C., & Davies, J. (2002). Nurses' job satisfaction, organizational commitment, and career intent. *Journal of Nursing Administration, 32*(5), 250-263.

Inoue, T., & Furuhashi, T. (2003). A proposal of combined method of evolutionary algorithm and heuristic for nurse scheduling support system. *IEEE Transactions on Industrial electronics, 50*, 833-841.

Isken, M. (2004). An implicit tour scheduling model with applications in healthcare. *Annals of Operations Research, 128*(1-4), 91-109.

Jan, A., Yamamoto, M. & Ohuchi, A. (2000). Evolutionary algorithms for nurse scheduling problem. *In Proceedings of IEEE Congress on Evolutionary Computation.* 1, 196-203.

Jaumard, B., Semet, F., & Vovor, T. (1998). A generalized linear programming model for nurse scheduling. *European Journal of Operational Research, 107*, 1-18.

Johari, H., Shamsuddin, F., Idris, N., & Hussin, A. (2013). Medication errors among nurses in government hospital. *Journal of Nursing and Health Science, 1*(2), 18-23.

Judith, A. O. (2006). The global nursing shortage: an overview of issues and actions. *Policy Politics Nursing Practice, 7*(3), 34-39.

Kahar, M. N. M., & Kendall, G. (2010). The examination timetabling problem at university Malaysia Pahang: comparison of constructive heuristic with an existing software solution. *European Journal of Operational Research*, 207, 557-565.

Kakas, A. (2000). *Research report 1997-1999.* Nicosia, Cyprus: Department of Computer Science, University of Cyprus.

Kalisch, B. J., & Aebersold, M. (2010). Interruptions and multitasking in nursing care. *The Joint Commission Journal on Quality and Patient Safety. 36*(3), 126-132.

Kane-Urrabazo, C. (2006). Management's role in shaping organizational culture. *Journal of Nursing Management*, *14*, 188-194.

Kazimipour, B., Li, X., & Qin, A. Q. (2014). A review of population initialization techniques for evolutionary algorithms. In *IEEE Congress on Evolutionary Computation (CEC)*, 2585-2592.

Kelemci, O., & Uyar, A. S. (2007). Application of a genetic algorithm to a real world nurse rostering problem instance. In *Proceedings of the 9th International Conference on Enterprise Information Systems ICEIS (2), Madeira, Portugal*, 474-477.

Kelemen, A., Franklin, S., & Liang, Y. L. (2005). Constraint satisfaction in "conscious" software agents- a practical application. *Applied Artificial Intelligence*, *19*, 491-514.

Kellegöz, T., Toklu, B., & Wilson, J. (2008). Comparing efficiencies of genetic crossover operators for one machine total weighted tardiness problem. *Applied Mathematics and Computation*, *199*(2), 590-598.

Kessler, C., & Manta, V. (1990). Specificity of restriction endonucleases and DNA modification methyltransferases- a review (Edition 3). *Gene*, *92*(1), 1-240.

Khaji, E., & Mohammadi, A. S. (2014). A heuristic method to generate better initial population for evolutionary methods. *Neural and Evolutionary Computing*, 1406-4518.

Kim, S. J., Ko, Y. W., Uhmn, S. Y., & Kim, J. (2014). A strategy to improve performance of genetic algorithm for nurse scheduling problem. *International Journal of Software Engineering and Its Applications*, *8*(1), 53-62.

Kirkpatrick, S, Gelatt, C. D., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science. 220*(4598), 671-680.

Korošec, P., Bole, U., & Papa, G. (2013). A multi-objective approach to the application of real-world production scheduling. *Expert Systems with Applications*, *40*(15), 5839-5853.

Kumar, R., Tyagi, S., & Sharma, M. (2013). Memetic algorithm: hybridization of hill climbing with selection operator. *International Journal of Soft Computing and Engineering, 3*(2), 2231-2307.

Kumara, B. T. G. S., & Perera, A. A. I. (2011). Automated system for nurse scheduling using graph coloring. *Indian Journal of Computer Science and Engineering (IJCSE)*, *2*(3), 476-485.

Lacomme, P., Prins, C., & Ramdane-Chérif, W. (2005). Evolutionary algorithms for periodic arc routing problems. *European Journal of Operational Research, 165*, 535-553.

Landro, L. (2008). Hospitals move to reduce risk of night shift. *The Wall Street Journal.* Retrieved on August 16, 2010 from http://www.wsj.com/articles/SB121193074899024387

Lederberg, S., & Meselson, M. (1964). Degradation of non-replicating bacteriophage DNA in non-accepting cells. *Journal of molecular biology*, *8*(5), 623-628.

Legg, S., & Hutter, M. (2005). Fitness uniform deletion: a simple way to preserve diversity. In *Proc. of the Genetic and Evolutionary Computation Conference (GECCO).* USA, 1271-1278.

Legg, S., Hutter, M., & Kumar, A. (2004). Tournament versus fitness uniform selection. *In proceeding of the 2004 Congress on Evolutionary Computation (CEC'04)*. Portland, OR: IEEE, 2144-2151.

Lewis, R. M. R., & Paechter, B. (2004). New crossover operators for timetabling with evolutionary algorithms. In *Lofti A (ed) 5th International Conference on Recent Advances in Soft Computing RASC2004*, Nottingham, UK. 5, 189-195.

Li, H. B., Lim, A., & Rodrigues, B. (2003). A hybrid ai approach for nurse rostering problem. In *Proceedings of the 2003 ACM Symposium on Applied Computing (SAC)*, Melbourne. IEEE Press, 730-735.

Lim, H. T., & Ramli, R. (2010). Recent advancements of nurse scheduling models and a potential path. In *Proceedings of the 6th IMT-GT Conference on Mathematics, Statistics and its Applications (ICMSA2010)*, 395-409.

Lin, C. C. (2009). An evolutionary algorithm with non-random initial population for path planning of manipulators. *Next-Generation Applied Intelligence*, *5579*, 193-201.

Lin, E., Herrmann, J. W., & Vieira, G. E. (2003). Rescheduling manufacturing systems: a framework of strategies, policies, and methods. *Journal of scheduling*, *6*, 39-62.

Lin, L., Gen, M., & Wang, X. G. (2009). Integrated multistage logistics network design by using hybrid evolutionary algorithm. *Computers & Industrial Engineering*, 56(3), 854-873.

Lin, W. Y. (2010). A ga–de hybrid evolutionary algorithm for path synthesis of four-bar linkage. *Mechanism and Machine Theory*, *45*(8), 1096-1107.

Lin, W. Y., Lee, W. Y., & Hong, T. P. (2003). Adapting crossover and mutation rates in genetic algorithms. *Journal of Information Science and Engineering*, *19*, 889-903.

Lozano, M., Herrera, F., Krasnogor, N. & Molina, D. (2004). Real-coded memetic algorithms with crossover hill-climbing. *Evolutionary Computation*, *12*(3), 273-302.

Lu, Z., & Hao, J. K. (2012). Adaptive neighborhood search for nurse rostering. *European Journal of Operational Research*, *218*(3), 865-876.

Maaranen, H., Miettinen, K., & Makela, M. M. (2004). Quasi-random initial population for genetic algorithms. *Computers & Mathematics with Applications*, *47*(12), 1885-1895.

Maenhout, B., & Vanhoucke, M. (2008a). Comparison and hybridization of crossover operators for the nurse scheduling problem. *Annals of Operations Research*, *159*(1), 333-353.

Maenhout, B., & Vanhoucke, M. (2008b). *Integrating the nurse staffing decision and the shift scheduling decision: case study and policy analysis* (Working paper 08/497). Ghent University.

Maenhout, B., & Vanhoucke, M. (2010). A hybrid scatter search heuristic for personalized crew rosteriing in the airline industry. *European Journal of Operational Research*, *206*, 155-167.

Maenhout, B., & Vanhoucke, M. (2011). An evolutionary approach for the nurse rerostering problem. *Journal of Computers &Operations Research*, *38*, 1400-1411.

Maenhout, B., & Vanhoucke, M. (2013). Reconstructing nurse schedules: computational insights in the problem size parameters. *Omega*, *41*, 903-918.

Mahfound, S. W. (2000). Boltzmann selection. In *Evolutionary Computation I: Basic Algorithms and Operators (Back, T., Fogel, D. B. & Michalewicz, Z., eds.).* 195-200, Bristol: Institute of Physics Publishing.

Malaysia Health System Review. (2013). *Health systems in transition*. *3*(1), 1-104. Retrieved on March 5, 2014 from http://www.wpro.who.int/asia_pacific_observatory/hits/series/Malaysia_Health_Systems_Review2013.pdf

Malim, B., & Wessberg, J. (2010). A memetic algorithm for selection of 3D clustered featured with application in neuroscience. In *Proceedings of International Conference on Pattern Recognition IEEE*, 1076-1079.

Marichelvam, M. K. (2012). An improved hybrid cuckoo search (ihcs) metaheuristics algorithm for permutation flow shop scheduling problems. *International Journal of Bio-Inspired Computation*, *4*(4), 200-205.

Marichelvam, M. K., Prabaharan, T., & Yang, X. S. (2014). Improved cuckoo search algorithm for hybrid flow shop scheduling problems to minimize makespan. *Applied Soft Computing*, *19*, 93-101.

Martínez-Estudillo, A., Hervás-Martínez, C., Martínez-Estudillo, F., & García-Pedrajas, N. (2006). Hybridization of evolutionary algorithms and local search by means of a clustering method. *IEEE Transactions on Systems, Man, and Cybernetics: Part B - Cybernetics*, *36*(3), 534-545.

Mashwani, W. K., & Salhi, A. (2012). A decomposition-based hybrid multiobjective evolutionary algorithm with dynamic resource allocation. *Applied Soft Computing*, 12(9), 2765-2780.

Mathes, E. W. (1981). Maslow's hierarchy of needs as a guide for living. *Journal of Humanistic Psychology*, *21*(4), 69-72.

Matthews, C. H. (2005). Using linear programming to minimize the cost of nurse personnel. *Journal of Health Care Finance*, *32*(1), 37-49.

Maziah, A. M., Wichaikhum, O., & Nantsupawat, R. (2012). Nursing practice environment and patient outcomes in university hospitals in malaysia. *Health and the Environment Journal, 3*(1), 16-26.

McCoy, S. P., & Aamodt, M. G. (2010). A comparison of law enforcement divorce rates with those of other occupations. *Journal of Police and Criminal Psychology, 25*, 1-16.

McEachen, I., & Keogh, J. (2007). *Nurse management demystified.* New York: McGraw-Hill.

McMenamin, T. M. (2010). Issues in labor statistics: illness-related work absences during flu season. *US Bureau of Labor Statistics*, 1-4.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equation of state calculations by fast computing machines. *The Journal of Chemical Physics*, *21*(6), 1087.

Michalewicz, Z., & Fogel, D. B. (2002). *How to solve it: modern heuristics*. New York: Springer.

Miki, M., Hiroyasu, T., Yoshida, J. I., & Ohmukai, I. (2000). New crossover scheme for parallel distributed genetic algorithms. In *Proceeding of the IASTED International Conference, Parallel and Distributed Computing And Systems*, 1, 145-150.

Ministry of Health Malaysia. (2004). Retrieved on September 19, 2008 from http://www.moh.gov.my/MohPortal/index.jsp

Ministry of Health Malaysia. (2007). *Malaysia's health*. Retrieved on June 10, 2011 from http://www.moh.gov.my/images/gallery/publications/mh/Malaysia%20Health%202007-2.pdf

Ministry of Health Malaysia. (2008). *Annual report 2008*. Retrieved on March 4, 2014 from http://www.moh.gov.my/images/gallery/publications/md/ar/2008-4.pdf

Ministry of Health Malaysia. (2012). *Health facts 2012. Health Informatics Centre, Planning and Development Division*. Retrieved on August 16, 2013 from http://www.moh.gov.my/images/gallery/stats/heal_fact/health_fact_2012_page_by_page.pdf

Ministry of Health Malaysia. (2013). *Health facts*. Retrieved on March 3, 2014 from http://www.youblisher.com/p/723128-MOH-Malaysia/

Missouri State Board of Nursing. (2008). *Health facts 2008*. Retrieved on March 23, 2009 from http://www.moh.gov.my/opencms/export/sites/default/moh/download/health_fact_2008_page_by_page

Mitchell, M. (1996). *An introduction to genetic algorithm. Complex adaptive systems*. Cambridge, MA, US: The MIT Press. viii, 205.

Montgomery, J., & Chen, S. (2010). An analysis of the operation of differential evolution at high and low crossover rates. In *IEEE Congress on Evolutionary Computation (CEC)*, 1-8.

Morton, T. E., & Pentico, D. W. (1993). *Heuristic scheduling systems with applications to production systems and project management*. Canada: John Wiley.

Moz, M., & Pato, M. V. (2003). An integer multicommodity flow model applied to the rerostering of nurse schedules. *Annals of Operations Research*, *199*, 285-301.

Moz, M., & Pato, M. V. (2004). Solving the problem of rerostering nurse schedules with hard constraints: new multicommodity flow models. *Annals of Operations Research, 128*, 179-197.

Moz, M., & Pato, M. V. (2007). A genetic algorithm approach to a nurse rerostering problem. *Computers and Operations Research*, *34*, 667–691.

Muntz, A. H., & Wang, K. (1990). Workload model specifications and adaptive scheduling of semi-hard real-time controls. In *Proceedings of the First International Conference on Systems Integration,* IEEE, 403-414.

Murate, T. & Ishibuchi, H. (1994). Performance evaluation of genetic algorithms for flowshop scheduling problems. In *International Conference on Evolutionary Computation*. 812-817.

Narasimhan, R. (1996). An algorithm for single shift scheduling of hierarchical workforce. *European journal of Operational Resea*rch, *96*, 113-121.

Needleman, J., Buerhaus, P., Pankratz, V. S., Leibson, C. L., Stevens, S. R., & Harris, M. (2011). Nurse staffing and inpatient hospital mortality. *Journal of Medicine*, *364*, 1037-1045.

Nonobe, K., & Ibaraki, T. (1998). A tabu search approach to the constraint satisfaction problem as a general problem solver. *European Journal of Operational Research*, *106*, 599-623.

Nursing and Midwifery Office. (2012). *Work arrangements: principles of best practice rostering- queensland health guidelines* (3rd ed.). Queensland: Nursing and Midwifery Office. Retrieved from http://www.qnu.org.au/__data/assets/pdf_file/0019/381007/20121119-NM_Rostering_-Guidelines-Final.pdf

Ouaarab, A., Ahiod, B., & Yang, X. S. (2014). Discrete cuckoo search algorithm for the traveling salesman problem. *Neural Computing and Applications*, *24*(7-8), 1659-1669.

Oughalime, A., Ismail, W. R., & Liong, C. Y. (2008). A tabu search to the nurse scheduling problem. In *Prooceeding of International Symposium on Information Technology(ITSim2008), Kuala Lumpur, Malaysia*, *1*, 1-7.

Özcan, E. (2005). Memetic algorithms for nurse rostering. In *Proceeding of the 20th International Conference on Computer and Information Sciences*. Istanbul, Turkey.

Özcan, E. (2006). An empirical investigation on memes, self-generation and nurse rostering. In *Proceeding of the 6th International Conference on the Practice and Theory of Automated Timetabling*, 246-263.

Pamela, M. C. (2007). 7/70: an innovative scheduling model for recruitment and retention success. *Magnet Conference of ANCC American Nurses Credentialing Center.* Retrieved on December 16, 2009 from http://hdl.handle.net/10755/182861.

Pendharkar, P. C., & Rodger, J. A. (2004). An empirical study of impact of crossover operators on the performance of non-binary genetic algorithm based neural approaches for classification. *Journal of Computers & Operations Research*, *31*, 481-498.

Peters, E., Matta, R., & Boe, W. (2007). Short-term work scheduling with job assignment flexibility for a multi-fleet transport system. *European Journal of Operational Research*, *180*, 82-89.

Pierce, N. A., & Winfree, E. (2002). Protein design is np-hard. *Protein Engineering, 15*(10), 779-782.

Pinedo, M. (2002). *Scheduling: theory, algorithms and* system (2nd Edition). New Jersey: Prentic-Hall.

Pingoud, A., Alves, J., & Geiger, R. (1993). Restriction enzymes. In *Enzymes of Molecular Biology*,107-200. Humana Press.

Porumbel, D. C., Hao, J. K. & Kuntz, P. (2009). Diversity control and multi-parent recombination for evolutionary graph coloring algorithms. In *Evolutionary Computation in Combinatorial Optimization*, 5482 of *LNCS*, 121-132.

Potthoff, D., Huisman, D., & Desaulniers, G. (2010). Column generation with dynamic duty selection for railway crew rescheduling. *Transportation Science*, *44*, 493-505.

Prodhon, C. (2011). A hybrid evolutionary algorithm for the periodic location-routing problem. *European Journal of Operational Research, 210*(2), 204-212.

Punnakitikashem, P. (2007). *Integrated nurse staffing and assignment under uncertainty*. The University of Texas at Arlington.

Punnakitikashem, P., Rosenberger, J. M., Behan, D. F. B., Baker, R. L., & Goss, K. (2006). An optimization-based prototype for nurse assignment. In *Proc. of the 7th Asia Pacific Industrial Engineering and Management Systems Conference*, Bangkok, Thailand, 1080-1087.

Querstret, D., & Cropley, M. (2011). Why nurses need to unwind from work. *Nursing Times*, *107*(10), 14-17. Retrieved on January 10, 2014 from http://www.nursingtimes.net/Journals/2014/08/04/z/z/o/110315Why-nurses-need-to-unwind-from-work.pdf

Raghuwanshi, M. M. & Kakde, O, G. (2006). Genetic algorithm with species and sexual selection. In *Proceedings of the IEEE Conference on Cybernetics and Intelligent Systems (CIS)*. Piscataway, NJ. 1-8.

Rahnamayan, S., Tizhoosh, H. R., & Salama, M. M. A. (2007). A novel population initialization method for accelerating evolutionary algorithms. *Computers & Mathematics with Applications*, *53*(10), 1605-1614.

Ramli, R. (2004). *An evolutionary algorithm for nurse scheduling problem with circadian rhythms*. Unpublished Ph.D. Thesis. Universiti Sains Malaysia.

Razali, N. M., & Geraghty, J. (2011). Genetic algorithm performance with different selection strategies in solving tsp. In *Proceeding of the World Congress on Engineering WCE, London, UK.* 2.

Reeves, C. (1996). Hybrid genetic algorithms for bin-packing and related problems. *Annals of Operations Research.* 63, 371-396.

Reeves, C. R. (1993). Genetic algorithms. In *Modern heuristic techniques for combinatorial problems*, Oxford: Blackwell Scientific Press. 151-190.

Rezanova, N.J., & Ryan, D. M. (2010). The train driver recovery problem - a set partitioning based model and solution method. *Computers and Operations Research*, *37*, 845-856.

Riot, R. (2012). *Riot discourages nurses from working oversea: borneo post.* Retrieved on April 21, 2013 from http://www.theborneopost.com/2012/03/04/riot-discourages-nurses-from-working-overseas/

Roberts, R. J., & Murray, K. (1976). Restriction endonuclease. *Critical Reviews in Biochemistry and Molecular Biology*, *4*(2), 123-164.

Rocha, M., Oliveira, J. F., & Carravilla, M. A. (2013). Cyclic staff scheduling: optimization models for real-life problems. *Journal of Scheduling*, *16*, 231-242.

Rodrigues, D., Pereira, L. A. M., Almeida, T. N. S., Papa, J. P., Souza, A. N., Ramos, C. C. O., & Yang, X. S. (2013). A binary cuckoo search algorithm for feature selection. *In Proceedings of IEEE International Symposium on Circuits and Systems*, 465-468.

Rogalska, M., Bożejko, W., & Hejducki, Z. (2008). Time/cost optimization using hybrid evolutionary algorithm in construction project scheduling. *Automation in Construction*, *18*(1), 24-31.

Rosenberger, J. M., Johnson, E. L., & Nemhauser, G. L. (2003). Rerouting aircraft for airline recovery. *Transportation Science*, *37*(4), 408-421.

Salcedo-Sanz, S. (2009). A survey of repair methods used as constraint handling techniques in evolutionary algorithms. *Computer Science Reviews*, *3*, 175-192.

Shahriari, M., Shamali, M., & Yazdannik, A. (2014). The relationship between fixed and rotating shifts with job burnout in nurses working in critical care areas. *Iranian Journal of Nursing and Midwifery Research, 19*(4), 360-365.

Sharma, A., & Mehta, A. (2013). Review paper of various selection methods in genetic algorithm. *International Journal of Advanced Research In Computer Science And Software Engineering*, *3*(7), 1476-1479.

Shirey, M., Ebright, P., & McDaniel, A. (2008). Sleepless in america: nurse managers cope with stress and complexity. *The Journal of Nursing Administration*, *38*(3), 125-131.

Singh, S., Kurmi, J., & Tiwari, S. P. (2015). A hybrid genetic and cuckoo search algorithm for job scheduling. *International Journal of Scientific and Research Publications*, *5*(6), 1-4.

Sorensen, K., & Sevaux, M. (2006). Ma|pm: memetic algorithms with population management. *Computers & Operations Research*, 33, 1214-1225.

Soubeiga, E., (2003). *Development and application of hyperheuristics to personnel scheduling*. The University of Nottingham.

Srinivas, M., & Patnail, L. M. (1994). Adaptive probabilities of crossover and mutation in genetic algorithms. *IEEE Transactions On Systems, Man And Cybernetics*, *24*(4), 656-667.

Stephenson, J. (2014). NHS must value its nursing staff, warns report. *Nursing Times*. Retrieved on December 13, 2014 from http://www.nursingtimes.net/nursing-practice/specialisms/management/nhs-must-value-its-nursing-staff-warns-report/5067213.article

Stimpfel, A. W., Sloane, D. M., & Aiken, L. H. (2102). The longer the shifts for hospital nurses, the higher the levels of burnout and patient dissatisfaction. *Health Affairs (Project Hope)*, *31*(11), 2501-2509. doi:10.1377/hlthaff.2011.1377

Sudholt, D. (2009). The impact of parameterization in memetic evolutionary algorithms. *Theoretical Computer Science*, *410*, 2511-2528.

Suman, B., & Kumar, P. (2006). A survey of simulated annealing as a tool for single and multiobjective optimization. *Operational Research Society*, *57*, 1143-1160.

Syswerda, G. (1996). Schedule optimization using genetic algorithms. In *Handbook of Genetic Algorithms*. London: International Thomson Computer Press. 335-349.

Tang, F. I., Sheu, S. J., Yu, S., Wei, I, L., & Chen, C. H. (2007). Nurses relate the contributing factors involved in medication errors. *Journal of Clinical Nursing*, *16*(3), 447-457.

Tang, W. M., & Ghani, M. F. A. (2012). Job satisfaction among the nurse educators in the klang valley, Malaysia. *International Journal of Nursing Science*, 2(4), 29-33.

Taunton, R., Boyle, D., Woods, C., Hansen, H., & Bott, M. (1997). Manager leadership and retention of hospital staff nurses. *Western Journal of Nursing Research*,*19*(2), 205-226.

Temby, L., Vamplew, P., & Berry, A. (2005). *Accelerating real-valued genetic algorithms using mutation-with-momentum*. Hobart: University of Tasmania.

The Point of Care Foundation. (2014). How to engage staff in the nhs and why it matters: the state of play. *Staff report 2014*. Retrieved on December 20, 2014 from http://www.pointofcarefoundation.org.uk/Downloads/Staff-Report-2014.pdf

The Press Association. (2011). Junior staff more likely to take sick leave. *Nursing Times*. Retrieved on September 23, 2012 from http://www.nursingtimes.net/nursing-practice/clinical-zones/public-health/junior-staff-more-likely-to-take-sick-leave/5025534.article

Thengvall, B. G., Bard, J. F., & Yu, G. (2000). Balancing user preferences for aircraft schedule recovery during irregular operations. *IIE Transactions*. *32*(3), 181-193.

Thompson, G. M. (1998). Labor scheduling, part 1. *Cornell Hotel and Restaurant Adm Q*, *39*(5), 22-31.

Thompson, G. M. (1999). Labor scheduling, part 3. *Cornell Hotel and Restaurant Adm Q*, *40*(1), 86-107.

Thompson, G. M. (2007). Solving the multi-objective nurse scheduling problem with a weighted cost function. *Ann Oper Res*. 155, 279-288.

Till J., Sand G., Engell, S., Emmerich, M. T. M., & Schönemann, L. (2005), A hybrid algorithm for solving two-stage stochastic integer problems by combining evolutionary algorithms and mathematical programming methods. In: Puigjaner L. (Ed.) *Proceedings European Symposium Computer Aided Process Engineering (ESCAPE-15)*, Computer-aided Chemical Engineering, Amsterdam: Elsevier, 187-192.

Ting, C. K., & Buning, H. K. (2003). A mating strategy for multi-parent genetic algorithms by integrating tabu search. In *The 2003 IEEE Congress on Evolutionary Computation (CEC'03)*, 2, 1259-1266.

Toledo, C. F. M., da Silva Arantes, M., De Oliveira, R. R. R., & Almada-Lobo, B. (2013). Glass container production scheduling through hybrid multi-population based evolutionary algorithm. *Applied Soft Computing*, *13*(3), 1352-1364.

Tometzki, T., & Engell, S. (2009). Hybrid evolutionary optimization of two-stage stochastic integer programming problems: an empirical investigation. *Journal of Evolutionary computation*, *17*(4), 511-526.

Topaloglu, S., & Ozkarahan, I. (2004). An implicit goal programming model for the tour scheduling problem considering the employee work preferences. *Annals of Operations Research*, *128*(1-4), 135-158.

Tourangeau, A. E., & Cranley, L. A. (2006). Nurse intention to remain employed: Understanding and strengthening determinants. *Journal of Advanced Nursing*, *55*, 497-509.

Tsai, C. C., & Li, S. H. (2009). A two-stage modeling with genetic algorithms for the nurse scheduling problem. *Expert Systems with Applications*, *36*(5), 9506-9512.

Tuba, M., Subotic, M., & Stanarevic, N. (2011). Modified cuckoo search algorithm for unconstrained optimization problems. In *proceeding of the European computing Conference,* 263-268.

Tušar, T., & Filipič, B. (2007). Differential evolution versus genetic algorithms in multiobjective optimization. In *Proceedings of the Fourth International Conference on Evolutionary Multi-Criterion Optimization*, *4403*, 257-271.

U.S. Bureau of Labor Statistics. (2008). Workdays for people in healthcare occupations. *Issues in Labor Statistics.* Retrieved on January 4, 2011 from http://www.bls.gov/opub/btn/archive/workdays-for-people-in-healthcare-occupations-pdf.pdf

Urnov, F. D., Rebar, E. J., Holmes, M. C., Zhang, H. S., & Gregory, P. D. (2010). Genome editing with engineered zinc finger nucleases. *Nature Reviews Genetics*, *11*(9), 636-646.

Urselmann, M., Emmerich, M. T. M., Till, J., Sand, G., & Engell, S. (2007). Design of problem-specific evolutionary algorithm/mixed-integer programming hybrids: two-stage stochastic integer programming applied to chemical batch scheduling, *Engineering Optimization*, *39*(5), 529-549.

Vahey, D. C., Aiken, L. H., Sloane, D. M., Clarke, S. P., & Vargas, D. (2004). Nurse burnout and patient satisfaction. *Med Care*, *42*(2), 1157–1166.

Valouxis, C., & Housos, E. (2000). Hybrid optimization techniques for the workshift and rest assignment of nursing personnel. *Artificial Intelligence in Medicine*, *20*, 155-175.

Valouxis, C., Gogos, C., Goulas, G., Alefragis, P., & Housos, E. (2012). A systematic two phase approach for the nurse rostering problem. European *Journal of Operational Research*, *219*, 425-433.

Van den Bergh, J., Beliën, J., De Bruecker, P., Demeulemeester, E., & De Boeck, L. (2013). Personnel scheduling: A literature review. *European Journal of Operational Research*, *226*(3), 367-385.

Veen, E. V. D., Hans, E. W., Post, G., & Veltman, B. (2012). Shift rostering using decomposition: assign weekend shifts first. *Journal of Scheduling*, 1-17.

Veerapen, N., Maturana, J., & Saubion, F. (2012). An exploration-exploitation compromise-based adaptive operator selection for local search. In *Proceedings of the 2012 Genetic and Evolutionary Computation Conference GECCO*, 1277-1284.

Vieira, G. E., Herrmann, J. W., & Lin, E. (2000). Predicting the performance of rescheduling strategies for parallel machine systems. *Journal of Manufacturing Systems*, *19*(4), 256-266.

Vila, B., Morrison, G. B., & Kenney, D. (2002). Improving shift scheduling and work-hour policies and practices to increase police officer performance, health, and safety. *Police Quarterly*, *5*(1), 4-24.

Vincze, T., Posfai, J., & Roberts, R. J. (2003). NEBcutter: a program to cleave DNA with restriction enzymes. *Nucleic acids research*, *31*(13), 3688-3691.

Wagner, S., Affenzeller, M., & Schragl, D. (2004). Trapes and dangers when modeling problems for genetic algorithms. *Cybernetics and Systems*, 79-84.

Wang, C. W., Sun, L. M., Jin, M. H., Fu, C. J., Liu, L., Chan, C. H., & Kao, C. Y. (2007). A genetic algorithm for resident physician scheduling problem. *In Proceeding of the 9th annual conference on Genetic and evolutionary computation*, 2203-2210.

Wang, L., & Li, L. P. (2010). An effective hybrid quantum-inspired evolutionary algorithm for parameter estimation of chaotic systems. *Expert Systems with Applications*, *37*(2), 1279-1285.

Warner, D. M. (1976). Scheduling nursing personnel according to nursing preferences: a mathematical programming approach. *Operations Research*, *24*(5):842-856.

Weide, O., Ryan, D., & Ehrgott, M. (2010). An iterative approach to robust and integrated aircraft rounting and crew scheduling. *Computers and Operations Research, 37*, 833-844.

Weil, G., Heus, K., Francois, P., & Poujade, M. (1995). Constraint programming for nurse scheduling. *IEEE Engineering in Medicine and Biology*, *14*(4), 417-422.

Whitley, D. (2001). An overview of evolutionary algorithms: practical issues and common pitfalls. *Information and Software Technology*, *43*, 817-831.

Williams, C. (2008). Work-life balance of shift workers. *Perspectives on Labour and Income*, *9*(8), 5-16.

Winstanley, G. (2004). Distributed and devolved work allocating planning. *Applied Artificial Intelligence*, *18*, 97-115.

Wren, A. (1996). Scheduling, timetabling and rostering-a special relationship? In *The Practice and theory of Automated Timetabling: Selected Papers from the 1st International Conference, Lecture Notes in Computer Science 1153 (Ross, P. & Burke, E. eds.).* Berlin: Springer-Verlag, 47-75.

Wright, P. D., Bretthauer, K. M., & Cote, M. J. (2006). Reexamining the nurse scheduling problem: Staffing ratios and nursing shortages. *Decision Sciences*, *37*, (1), 39-67.

Wu, T. H., Yeh, J. Y., & Lee, Y. M. (2015). A particle swarm optimization approach with refinement procedure for nurse rostering problem. *Journal of Computers and Operations Research*, *54*, 52-63. doi: 10.1016/j.cor.2014.08.016

Xiang, W. L., Ma, S. F., & An, M. Q. (2014). Habcde: a hybrid evolutionary algorithm based on artificial bee colony algorithm and differential evolution. *Applied Mathematics and Computation*, *238*, 370-386.

Yang, F. C., & Wu, W. T. (2012). A genetic algorithm-based method for creating impartial work schedules for nurses. *International Journal of Electronic Business Management*, *10*(3), 182.

Yang, X. S. (2014). *Nature-inspired optimization algorithms*. Elsevier, London

Yang, X. S., & Deb, S. (2009). Cuckoo search via levy flights. In *Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, India, 210-214.

Yang, X. S., & Deb, S. (2010). Engineering optimization by cuckoo search. *International Journal of Mathematical Modeling and Numerical Optimization*, *1*(4), 330-343.

Yang, X. S., & Deb, S. (2014). Cuckoo search: recent advances and applications. *Neural Computing and Applications*, *24*(1), 169-174.

Yano, C. (2005). Research. *IIE Transactions*, *37*(7), 50.

Yao, X., Liu, Y., & Lin, G. M. (1999). Evolutionary programming made faster. *IEEE Transactions on Evolutionary Computation*, *3*(2), 82-102.

Yat, G. C. W., & Hon, W. C. (2003). Nurse rostering using constraint programming and meta-level reasoning. In *Proceeding of the 16th International Conference on Developments in Applied Artificial Intelligence*, 712-721.

Yi, Y. (2005). *Application of finite-state machines in medical care scheduling*. Electrical and Computer Engineering, Duke University, Durham.

Younas, I., Kamrani, F., Moradi, F., Ayani, R., Schubert, J., & Håkansson, A. (2013). Solving battalion rescheduling problem using multi-objective genetic algorithms. In *AsiaSim 2013*, 93-104. Springer Berlin Heidelberg.

Zaharie, D. (2009). Influence of crossover on the behavior of differential evolution algorithms. *Journal of Applied Soft Computing*, *9*(3), 1126–1138.

Zhang, Q., Sun, J., & Tsang, E. (2005). An evolutionary algorithm with guided mutation for the maximum clique problem. *IEEE Transactions on Evolutionary Computation*, *9*(2), 192-200.

Zhang, W., Xu, W., & Gen, M. (2013). Multi-objective evolutionary algorithm with strong convergence of multi-area for assembly line balancing problem with worker capability. *Procedia Computer Science, 20,* 83-89. doi:10.1016/j.procs.2013.09.243

Zheng, H., Zhou, Y., & Guo, P. (2013). Hybrid genetic-cuckoo search algorithm for solving runway dependent aircraft landing problem. *Research Journal of Applied Sciences, Engineering and Technology*, *6*(12), 2136-2140.

Zhong , Y. W., & Yang, J. G. (2004). A hybrid genetic algorithm with lamarckian individual learning for tasks scheduling. *IEEE International Conference on Systems, Man and Cybernetics, Hague, Netherlands*, 3343-3348.

Zhong, J., Hu, X., Gu, M., & Zhang, J. (2005). Comparison of performance between different selection strategies on simple genetic algorithms. In *International Conference on Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce,* IEEE. 2, 1115-1121.

Appendix A

A random sample of a list of nurses' requested off day in a schedule

```
RO[1, 7];
RO[2,11];
RO[4,6];
RO[32,5];
RO[20,10];
RO[31,1];
RO[4,13];
RO[4,14];
RO[5,12];
RO[23,14];
RO[24,2];
RO[3,8];
RO[6,3];
RO[7,2];
RO[11,2];
RO[12,6];
RO[16,12];
RO[27,3];
RO[28,1];
RO[29,4];
RO[30,3];
RO[17,8];
RO[18,11];
RO[19,9];
RO[33,4];
RO[34,6];
RO[35,4];
RO[13,1];
RO[14,2];
RO[8,12];
RO[37,7];
RO[38,2];
RO[9,14];
RO[10,5];
RO[15,6];
RO[21,10];
RO[25,13];
RO[22,12];
RO[26,1];
RO[39,7];
RO[36,5];
%UNTITLED2 Summary of this function goes here
% Detailed explanation goes here
```

Appendix B

Output of One Way ANOVA Test for the comparison of eleven models

ONEWAY

DataSet0] D:\PhD infor- compaq\HT Presentation & Writeup\NSRP final output-30 data.sav

Descriptives

NSRPFitness

	N	Mean	Std. Deviation	Std. Error	95% Confidence Interval for Mean Lower Bound	95% Confidence Interval for Mean Upper Bound	Minimum	Maximum
1	5	4239.40	1099.463	491.695	2874.24	5604.56	3033	5044
2	1	4040.00	.	.	.	.	4040	4040
3	5	5850.80	1640.258	733.546	3814.15	7887.45	4062	8050
4	12	4634.75	1783.163	514.755	3501.78	5767.72	3040	9044
5	24	3454.63	1283.920	262.079	2912.47	3996.78	2033	6042
6	9	4828.22	1565.491	521.830	3624.88	6031.57	2046	7046
7	25	2997.36	1136.793	227.359	2528.11	3466.61	1041	6039
8	10	4749.50	1061.430	335.654	3990.20	5508.80	3041	6065
9	30	3205.67	747.898	136.547	2926.40	3484.94	2032	4047
10	10	4845.60	1555.749	491.971	3732.68	5958.52	2033	7049
11	30	2404.27	1066.963	194.800	2005.86	2802.68	1033	5034
Total	161	3575.50	1500.078	118.223	3342.02	3808.98	1033	9044

Test of Homogeneity of Variances

NSRPFitness

Levene Statistic	df1	df2	Sig.
2.043	9	150	.038

Groups with only one case are ignored in computing the test of homogeneity of variance for Fitness.

ANOVA

NSRPFitness

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	1.396E8	10	13977050.162	9.518	.000
Within Groups	2.203E8	150	1468447.244		
Total	3.598E8	160			

Appendix C

A sample of the best schedule with 1033 best fitness

Nurses		Weekly Days					Weekend		Weekly Days					Weekend	
		1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th	11th	12th	13th	14th
Senior	1	P	M	M	M	E	E	Q	M	M	E	E	U	N	N
	2	E	M	M	E	E	M	W	E	M	M	T	P	N	N
	3	N	N	N	O	O	W	P	T	M	E	E	M	N	N
	4	N	N	N	O	O	T	E	M	M	M	E	E	B	Q
	5	N	N	N	O	O	W	P	E	M	E	E	T	M	M
	6	M	E	T	N	N	N	O	O	P	M	E	M	W	E
	7	P	T	M	N	N	N	O	O	E	M	M	E	W	E
	8	U	M	M	N	N	N	O	O	E	E	M	B	W	E
	9	U	M	E	P	E	M	N	N	N	O	O	E	M	Q
	10	U	M	E	E	B	M	N	N	N	O	O	E	M	W
	11	M	T	P	E	M	M	N	N	N	O	O	E	E	W
	12	P	M	E	E	M	Q	M	E	U	N	N	N	O	O
	13	B	E	E	M	M	E	W	U	M	N	N	N	O	O
	14	M	B	M	E	E	W	E	M	U	N	N	N	O	O
Junior	15	M	M	M	E	M	Q	E	M	E	P	U	E	E	E
	16	P	M	M	E	E	W	M	M	E	E	M	T	M	E
	17	U	M	E	E	M	E	M	B	M	M	E	E	E	W
	18	P	E	E	M	E	W	M	M	E	E	T	M	M	M
	19	E	E	M	M	U	E	E	M	B	M	M	M	W	E
	20	N	N	N	O	O	W	M	E	M	B	U	E	E	M
	21	N	N	N	O	O	W	E	E	U	B	M	M	M	E
	22	N	N	N	O	O	W	E	E	E	M	U	B	M	M
	23	N	N	N	O	O	U	E	E	P	M	E	M	M	Q
	24	M	T	P	N	N	N	O	O	E	M	E	E	W	M
	25	U	E	M	N	N	N	O	O	M	E	E	M	B	W
	26	T	E	M	N	N	N	O	O	P	E	M	E	W	M
	27	M	P	T	N	N	N	O	O	E	M	M	E	W	E
	28	T	M	M	P	E	M	N	N	N	O	O	E	E	W
	29	E	M	E	T	P	M	N	N	N	O	O	M	E	W
	30	E	M	T	P	M	M	N	N	N	O	O	E	E	W
	31	T	M	E	P	M	M	N	N	N	O	O	E	E	W
	32	E	E	M	M	B	W	M	E	U	N	N	N	O	O
	33	M	M	E	B	E	E	W	U	M	N	N	N	O	O
	34	E	M	P	E	E	Q	M	M	U	N	N	N	O	O
	35	M	E	E	B	M	E	W	U	M	N	N	N	O	O
	36	E	E	E	M	B	W	E	M	U	M	E	M	N	N
	37	E	M	E	M	M	E	Q	M	M	E	U	P	N	N
	38	M	B	E	M	M	W	M	M	E	E	E	U	N	N
	39	P	E	M	M	M	E	Q	E	E	U	M	M	N	N

* The best schedule: 1033 fitness value

Appendix D

Pseudo code of Matlab for calculating fitness in NSRP

```
tic
tot_nurse=39;
no_sen=14;
no_day=14;
Dr=0.4;


pop1=f1(39,14,no_day);  pop1=f2(pop1);
pureTRO=0;
for l1=1:39
  for k1=1:14
     if pop1(l1,k1)=='R'
       pureTRO=pureTRO+1;
     end
  end
end
    pureTRO;
pop1=f3(pop1,no_day);
pop1=f4(pop1,no_day);
pop1=f5(pop1,no_day);
pop1=f6(pop1,no_day);

% calculate fitness for new chrom after some change
c1=reqfitm(pop1,no_day);
c2=reqfite(pop1,no_day);
c3=sm(pop1,no_day);
c4=se(pop1,no_day);
r1=rowfitono(pop1,no_day);
r2=rowfitmebalance(pop1,no_day);
r3=afteroe(pop1,no_day);
r4=rowfitwe(pop1,no_day);
r5=rowfitemnk(pop1,no_day);

col_fit(ind)=0;
reqfitm_sum(ind)=0;
reqfite_sum(ind)=0;
sm_sum(ind)=0;
se_sum(ind)=0;

for j=1:no_day
    reqfitm_sum(ind)=reqfitm_sum(ind)+c1(j,ind);
    reqfite_sum(ind)=reqfite_sum(ind)+c2(j,ind);
    sm_sum(ind)=sm_sum(ind)+c3(j,ind);
    se_sum(ind)=se_sum(ind)+c4(j,ind);
end
col_fit(ind)=reqfitm_sum(ind)+reqfite_sum(ind)+sm_sum(ind)+se_sum(in
d);

row_fit(ind)=0;
rowfitono_sum(ind)=0;
rowfitmebalance_sum(ind)=0;
afteroe_sum(ind)=0;
rowfitwe_sum(ind)=0;
rowfitemnk_sum(ind)=0;
```

```
for i=1:39
    rowfitono_sum(ind)=rowfitono_sum(ind)+r1(i,ind);
    rowfitmebalance_sum(ind)=rowfitmebalance_sum(ind)+r2(i,ind);
    afteroe_sum(ind)=afteroe_sum(ind) +r3(i,ind);
    rowfitwe_sum(ind)=rowfitwe_sum(ind) +r4(i,ind);
    rowfitemnk_sum(ind)=rowfitemnk_sum(ind) +r5(i,ind);
end
row_fit(ind)=rowfitono_sum(ind)+rowfitmebalance_sum(ind)+afteroe_sum
(ind)+rowfitwe_sum(ind)+rowfitemnk_sum(ind);


totRQTB(ind)=0;
for l1=1:39
  for k1=1:14
    if pop1(l1,k1,ind)=='Q' || pop1(l1,k1,ind)=='B'||
pop1(l1,k1,ind)=='T'|| pop1(l1,k1,ind)=='R'
       totRQTB(ind)=totRQTB(ind)+1;
    end
  end
end
RDisappv(ind)=pureTRO-totRQTB(ind);

xRTolerc_sum(ind)=0;
for l1=1:tot_nurse
    Rrow(l1,ind)=0;
    Offrow(l1,ind)=0;
    xRTolerc(l1,ind)=0;

        for k1=1:no_day

             if pop1(l1,k1,ind)=='R'
               Rrow(l1,ind)=Rrow(l1,ind)+1;
             end

             if pop1(l1,k1,ind)=='W' || pop1(l1,k1,ind)=='P'||
pop1(l1,k1,ind)=='U'|| pop1(l1,k1,ind)=='F'|| pop1(l1,k1,ind)=='Z'
               Offrow(l1,ind)=Offrow(l1,ind)+1;
             end
        end

        if Offrow(l1,ind)>0 && Rrow(l1,ind)>0
            if (Offrow(l1,ind)-Rrow(l1,ind))>=0
                xRTolerc(l1,ind)=Rrow(l1,ind);
            else
                xRTolerc(l1,ind)=Offrow(l1,ind);
            end
        end
end
    xRTolerc(:,ind);  % show 39 nurses' total of RO disapproved
    xRTolerc_sum(ind)=sum(xRTolerc(:,ind));  % xRTolerc_sum for 1
indv@ table


fitness(ind)=col_fit(ind)+row_fit(ind)+(xRTolerc_sum(n)*1)+(RDisappv
(n)*20); %%% rescheduling: +(no_cell*5)+(no_c*20);
% rowfitono_sum, rowfitmebalance_sum, afteroe_sum, rowfitwe_sum,
rowfitemnk_sum
% reqfitm_sum, reqfite_sum, sm_sum, se_sum

toc
```